Technische
Universität
Braunschweig

Towards Tactical Lane Change Behavior Planning for Automated Vehicles

Simon Ulbrich

Towards Tactical Lane Change Behavior Planning for Automated Vehicles

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines Doktors
der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von:	Simon Martin Ulbrich
aus:	Gehrden
eingereicht am:	26.06.2017
mündliche Prüfung am:	18.08.2017
1. Referent:	Professor Dr. Markus Maurer
2. Referent:	Professor Mohan M. Trivedi, PhD
3. Referent:	Professor Dr. Bernd Lichte
Druckjahr:	2018

Preface

This thesis is a result of my time at the Institute of Control Engineering at Technische Universiät Braunschweig working on behalf of Volkswagen Group Research.

I would like to thank Professor Dr. Markus Maurer as an advisor, mentor, and incessant supporter of me as an individual, this thesis, and my field of research. I am particularly grateful for being entrusted with various responsibilities without close guidance. Without those sometimes distracting, yet fascinating tasks I would never have found enough motivation to successfully complete my time as a PhD student. I thank Professor Mohan Trivedi for co-supervising my thesis. Thank you to Professor Dr. Bernd Lichte for his personal support, guidance for my research, and for always going the extra mile on my behalf.

Further, I thank Professor Dr. Thomas Form, Thomas Ruchatz, Dr. Arne Bartels, Dr. Lutz Junge, Dr. Michael Darms, Dr. Bernd Rössler, and Dr. Jens Langenberg for their managerial support at Volkswagen, personal and scientific feedback, and for providing me with the necessary resources to successfully undertake this thesis and the behavior planning in "Jack", the *Audi A7 piloted driving concept* vehicle.

Special thanks to my "K-EFFS function" colleagues at Volkswagen, Dr. Simon Großjohann, Dr. Christian Appelt, and Kai Homeier for becoming sparring partners, test drivers, bug reporters, bug fixers, code monkeys, and friends. Without your work, mine would not have been possible. Thanks to the perception team, in particular Dr. Thorsten Graf, Dr. Jens Spehr, Toni Heidenreich, Dr. Alexander Hans, Markus Köchy, Stefan Schrandt, Dr. Thorsten Bagdonat, and all the others: We "function" developers might sometimes have given you a rough time with our requirements, nagging, and complaining, but believe me, if we had to implement the perception, the car would still not be driving. Additionally, I would like to thank the support team, namely Oliver Klages, Andre Hördt, Christian Finsterbusch, Mayhar Mousavi, Michaela Steinhoff, Kathrien Symkenberg, Dr. Marco Langerwisch, Jens Krause, and everyone else for keeping our fleet of test vehicles going and for providing a toolset for simulation and testing.

From the Stadtpilot team I would like to thank Andreas Reschka, Jens Rieken, Martin Escher, Till Menzel, Gerrit Bagschik, Fabian Schuldt, Dr. Tobias Nothdurft, Johanna Rieke, Dr. Richard Matthaei, Falko Saust, Dr. Sebastian Ohl, Sven Böhme, Jan Timo Wendler, Tobias Scheide, and Dr. Jaebum Choi for their endless hours of developing code and features; often even in their free time. These things aside, thanks for the awesome time we had together. Moreover, I thank my other colleagues from the institute as well as in particular Dr. Veronika Krapf and Stefanie Scheffer for resolving whatever unsolvable issue. Last of all, thanks to all my students who contributed to my lectures, wrote theses under my supervision, or supported my work as student assistants.

From the Automotive Research Center Niedersachsen (NFF) I would like to thank Dr. Udo-Willi Kögler for his personal support and mentoring, Dr. Wolfgang Bartsch, and Dr. Kathrien Inderwisch for their hands-on help to get things done, and all professors on the board and from the "intelligent vehicles" division for their trust and support. Also, to those colleagues from other institutes, who I spent endless evenings writing project proposals or similar things, thanks for making it not only work but also fun.

Braunschweig, Monday 19th June 2017

Abstract

Automated driving within one lane is a fascinating experience. Yet, it is even more interesting to go a step ahead: Making automated lane changes without human driver interaction. This thesis presents a concept and implementation demonstrated in “Jack”, the *Audi A7 piloted driving concept* vehicle.

Given that automated driving is in the media every other day already, why is it still such a big issue to do tactical behavior planning for automated vehicles? It is one of the core areas where it is surprisingly obvious why humans are currently so much smarter than machines: Tactical driving behavior planning is a social task that requires cooperation, intention recognition, and complex situation assessment. Without complex cognitive capabilities in today’s automated vehicles, it is core of this thesis to find simple algorithms that *pretend* intelligence in behavior planning.

In fact, such behavior planning in automated driving is a constant trade-off between utility and risk: The vehicle has to balance value dimensions such as safety, legality, mobility, and additional aspects like creating user and third party satisfaction. This thesis provides a framework to boil down such abstract dimensions into a working implementation. Several of the foundations for this thesis were developed as part of the Stadtpilot project at TU Braunschweig.

While there has been plenty of research on concepts being tested in perfect, simulated worlds only, the approaches in this thesis have been implemented and evaluated in real world traffic with uncertain and imperfect sensor data. The implementation has been tested, tweaked, and used in “Jack” for more than 50 000 km of automated driving in everyday traffic.

Zusammenfassung

Automatisiertes Fahren innerhalb eines Fahrstreifens ist eine faszinierende Erfahrung. Noch spannender ist es jedoch noch einen Schritt weiter zu gehen: Auch Fahrstreifenwechsel automatisiert auszuführen, ohne Interaktion mit einem Menschen als Fahrer. In dieser Dissertation wird hierfür ein Konzept und dessen Umsetzung in „Jack" präsentiert, dem *Audi A7 piloted driving concept* Fahrzeug.

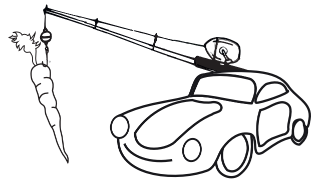
Automatisiertes Fahren ist aktuell in den Medien in aller Munde. Warum ist es dennoch eine große Herausforderung taktische Verhaltensplanung für automatisierte Fahrzeuge wirklich umzusetzen? Es ist einer der Kernbereiche, in denen offensichtlich wird, warum Menschen aktuell Maschinen im Straßenverkehr noch weitaus überlegen sind: Taktische Verhaltensplanung ist eine soziale Aufgabe, welche Kooperation, das Erkennen von Absichten und der Bewertung komplexer Situationen bedarf. Mangels wirklicher kognitiver Fähigkeiten in den heutigen automatisierten Fahrzeugen ist es Kern dieser Dissertation Algorithmen zu finden, welche zumindest den *Eindruck* intelligenter Verhaltensplanung erzeugen.

Eine solche Verhaltensplanung ist ein permanentes Abwägen von Nutzen und Risiken. Das Fahrzeug muss permanent Entscheidungen im Spannungsfeld zwischen Sicherheit, Legalität, Mobilität und weiten Aspekten wie Nutzerzufriedenheit und Zufriedenheit Dritter treffen. In dieser Dissertation wird ein Konzept entwickelt, um solche abstrakten Entscheidungsdimensionen in ein implementierbares Konzept herunterzubrechen. Viele Grundlagen dafür wurden im Rahmen des Stadtpilot Projekts der TU Braunschweig erarbeitet.

In vorausgehenden Arbeiten wurden bereits viele Ansätze entwickelt und auf Basis von perfekten, simulierten Daten evaluiert. Der in dieser Arbeit präsentierte Ansatz ist in der Lage mit unsicherheits- und fehlerbehafteten Messdaten umzugehen. Der Ansatz aus dieser Dissertation wurde in dem automatisiert fahrenden Fahrzeug „Jack" implementiert und bereits über 50 000 km im normalen Straßenverkehr genutzt und getestet.

Contents

1 Introduction



The Idea

Automated driving has been a vision for several decades. From its infancy until today, a long path of achievements has been accomplished. Automated driving within a lane is a fascinating experience. However, it is even more exciting to go a step ahead: Performing automated lane changes without prior clearance from a human driver or operator.

Contribution

This thesis presents a successfully implemented and publicly demonstrated approach for tactical behavior planning for lane changes in automated driving. It has been tested, tweaked and used for more than 50 000 km of automated driving.

1.1 Motivation

Excitement

Automated driving has been the focus of research for several decades. For the last 20 years, application oriented research has helped to narrow the gap between the vision and reality. Basic stabilization skills such as keeping a vehicle in a lane or basic distance keeping seems to be what journalists, potential customers or non-engineering persons are expecting from an automated vehicle. However, automatically executing tactical maneuvers is currently clearly among the delighting features (cf. Kano's theory of attractive quality, Kano cited by Hölzing, 2008, p. 77 ff.) stirring up a lot of excitement in favor of the technology.

Higher Levels of Automation

The SAE (2016), NHTSA (2013), and the BASt (Gasser et al., 2012) differentiated between different levels of automation. The higher the targeted level of automation is, the more tactical maneuver planning becomes a necessity for automated driving. In fact, tactical behavior planning is a stepping stone towards one day achieving fully automated vehicles.

Comfort and Safety

Furthermore, automating tactical behavior planning for lane changes may one day increase the comfort and safety of passengers in an automated vehicle. And even today, derived driver assistance systems only supporting a driver could help particular groups of drivers – e.g., elderly people – by simplifying tasks such as looking over the shoulder, enabling them to drive safer and with more comfort, or even allowing them to maintain mobility.

1.2 Context of this Thesis

Research towards tactical lane change behavior planning is conducted in both *academic* and *industry driven research.*

Academic Research

Academic research has contributed to formulating models for the theory of behavior in the fields of psychology, ethics, and sociology. Moreover, it has helped to find quantitative behavior models for how traffic or individual traffic participants behave. In the last few decades, automated driving has also become the focus of applied research. Several prototypical implementations of automated driving have been developed and refined. Groundbreaking research within Professor Dickmanns' group or for the DARPA Grand and Urban Challenges increased interest and expectations surrounding soon to be marketable automated vehicles.

Industry Driven Research

Industry driven research often follows rather a bottom-up approach: Moving on from advanced driver assistance systems and their constituents towards prototypes for automated vehicles. Either founded by car or information technology companies, several demonstrations of automated driving have been accomplished in the last decade. For some of them, it is vague and not clear what skills the automated vehicles really have regarding tactical lane change behavior planning.

Tactical Planning and Humans

Earlier contributions to automated driving often excluded tactical decision making from the skills of an automated vehicle. Tactical maneuvers were either to be confirmed by a human driver after a recommendation or were to be executed by a human driver entirely. This last step of handing an ultimate decision over to a human happens for good reason: The algorithms need to perform very well, to render a computer-decided maneuver execution feasible. Only a minimal rate of errors is acceptable, if maneuvers are directly executed. In fact, the solution presented in this thesis performed sufficiently well to allow even journalists to be seated in the driver's seat while driving automatically. Yet, the presented behavior planning is still monitored by a human. Hence, it is rather one stepping stone towards fully automated vehicles than its ultimate accomplishment.

1.3 Research Objective and Scientific Contribution

Objective

Goal of this thesis is to develop and evaluate a method for *best possible* tactical behavior planning for lane changes in automated driving. What does *best possible* imply? Behavior planning should be fast, consistent, provident, deterministic, and compliant with values. Overall, the algorithms should imitate[1] the behavior of a human driver, or better a human chauffeur, as well as possible.

Systemic

Among the major contributions for this thesis is a systemic approach to tactical behavior planning. Apart from algorithmic details, this thesis discusses broader implications associated with lane change behavior planning. In particular, the scientific contributions of this thesis are:

Scientific Contribution

- Definition of relevant terminology for research in tactical behavior planning in automated driving by defining the terms scene, situation, scenario, context, maneuver, behavior, intentions, skills, and abilities.
- Refinement of a functional system architecture for an automated vehicle such that tactical behavior planning can be incorporated into the overall structure of an automated vehicle's software design.
- Distinction of cooperative behavior tasks by cooperative skills and abilities as against communication and awareness channels.

[1] Sections 15.6.2 and 16.1 address aspects where deviations from human behavior are favorable.

- The identification of ten necessary scenarios of cooperative behavior in tactical lane change behavior planning for automated driving.
- A literature review on eligible methods and frameworks for tactical behavior planning as well as a literature review on how other teams have approached tactical lane change behavior planning.
- Definition of meta requirements for tactical lane change behavior planning: Rapidity, consistency, providentness, determinisim, and complying with values. Further for a system's design the specification of functional requirements, user interface requirements, useability requirements, and performance requirements.
- Identification and definition of decision-relevant context information for tactical behavior planning: Definition of necessary elements of a scene implementation, definition of necessary elements of a situation implementation, and the definition of necessary situation aspects for lane change planning.
- Identification of the value dimensions of safety, legality, mobility, user satisfaction, and third party satisfaction for tactical behavior planning. Defining a way to relate such abstract value dimensions towards a technical implementation.
- Demonstration how lane change behavior planning under uncertainty can be broken down into a measurement model, a situation prediction model, a reward model, and a planning core.
- A literature review on metrics for behavior planning performance evaluation.
- Demonstration of a way for a simulation-based performance evaluation by situation-based open-loop testing and scenario-based closed-loop testing.
- Demonstration of a maneuver-based performance evaluation in real traffic for three scenarios.
- Definition and evaluation of a metric for the situation prediction performance.
- Definition and evaluation of metrics for safety and mobility gains from tactical lane change planning.
- Definition and evaluation of a metric for the overall rate of correct and incorrect behavior decisions.
- Evaluation of the macroscopic performance by a test person study on how they judge the lane change behavior planning.
- Discussion of the limitations of the chosen approach and the identification of open issues, not addressed in this thesis.

This thesis is dominated by the two major aspects; on the one hand the world of academic models, on the other hand the challenge to find a working solution for real world driving with incorrect and incomplete information and immanent

Limitations

uncertainty.[2] This thesis should be understood as a contribution to bridge the gap. Yet it cannot provide any final solutions. To the author, it is rather purpose to open the field for future research than to provide indisputable answers.

1.4 Structure of this Thesis

This thesis is comprised of three parts, an introduction, and a conclusion. In the introduction, the context of this work is discussed and the goals and scientific contributions are highlighted.

Fundamentals

In the first part, the fundamentals for this thesis are laid out. First of all, a common terminology is established. An overall functional system architecture for an automated vehicle is presented and refined, particularly for tactical decision making. The often vague term of cooperative behavior will be substantiated and cooperative behavior scenarios for tactical lane change behavior planning will be identified. The section concludes with a literature review on the applied methods and publications focusing on addressing the same or similar applications as in this thesis.

Concepts & Implementations

The second part discusses the concepts and their implementations for tactical behavior planning for lane changes. First of all, the requirements for tactical decision making for lane changes are specified. Since lane change behavior planning requires context modeling, this part presents how a scene and situation description is aggregated from various perceived and interpreted pieces of information. Thereafter, the second part relates lane change behavior planning towards superordinate goals and values for an automated vehicle's behavior planning. Finally, the second part describes the lane change planning implementation with its constituting components, a measurement model, a situation prediction model, a reward and cost model, and a tactical planning core utilizing those models.

Metrics & Evaluations

The third part discusses metrics to quantify the performance in behavior planning and performs an evaluation of the developed algorithms based on those. Initially, this evaluation is performed in a simulation with ideal data. Thereafter, distinct maneuvers are evaluated in detail. Last of all, a macroscopic evaluation over several hundred maneuvers is performed.

This thesis concludes with a research outlook in Chapter 16.

[2] Special thanks to Prof. Markus Maurer for pointing out this central aspect.

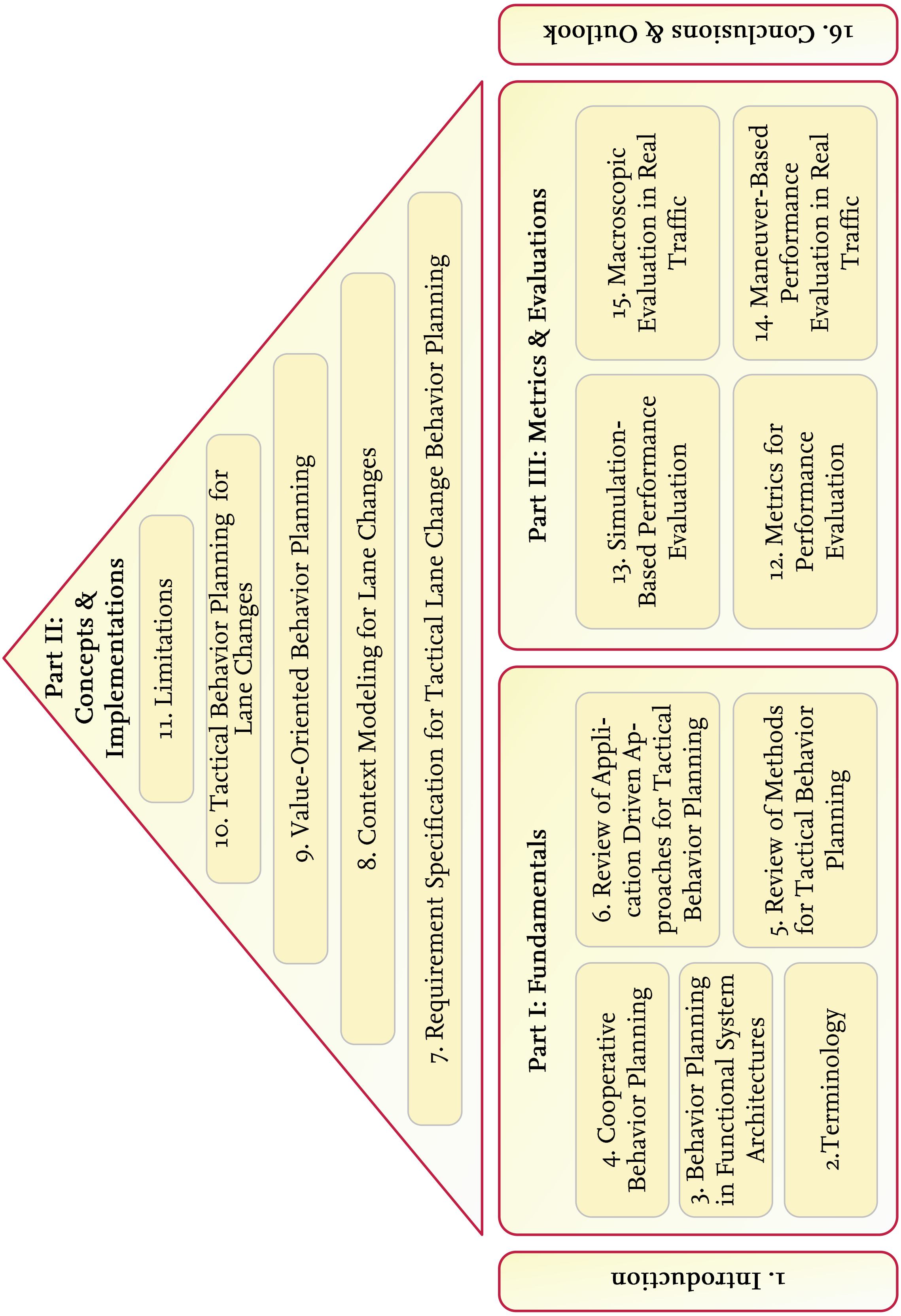
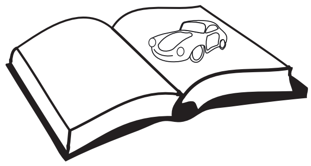


Figure 1.1: Structure of this Thesis

Part I

Fundamentals

2 Terminology



Scientific writing requires well-defined wording. This section provides definitions for essential terms for the remainder of this thesis.

After introducing the terms automated and assisted driving in section 2.1, the terms scene, situation, scenario, and context are clarified in section 2.3. The terminology is completed by defining the – for this thesis – central terms maneuver, behavior, intention, skills, and abilities in section 2.4 and uncertainty in section 2.5.

2.1 Automated and Assisted Driving

Levels of Automation

According to the definition of Gasser et al. (2012), the SAE (2016), and the NHTSA (2013), the term automated driving is used for vehicles that can – to some extent – drive themselves.

Partial and Conditional Automation

Gasser et al. (2012) and the SAE (2016) differentiate between assisted driving, partially automated driving, highly automated driving, and fully automated driving. Assisted driving only supports a driver by lateral *or* longitudinal control. The driver is in charge of supervising the system at all times. Partial automation supports the driver by longitudinal *and* lateral control, while the driver still has to supervise the system and to serve as a fallback level. The SAE (2016) introduces conditional automation for systems that take over longitudinal *and* lateral control and monitor its execution but are not responsible for bringing the system to a risk minimal state in case of a system limitation.

High Automation

The SAE (2016) differentiates highly automated driving from conditional automation by the fact that the system needs to find a risk minimal state in the case of a system limitation. Therefore, the system itself serves as a fallback level. In Gasser et al. (2012), highly automated driving relaxes the obligation of supervising the system at all times. Highly automated driving may be activated in certain conditions and domains. The system monitors itself and may only request a human operator to resume driving after a handover time. Within that handover time, the system has to make sure it handles driving in all conditions and performs fallback maneuvers to reach a risk minimal state. Hence, Gasser et al.'s (2012) and the SAE's (2016) definition of highly automated driving are similar up until the handover time has passed. Gasser et al. (2012) assume that a human serves as a fallback level after the handover time for highly automated driving. The SAE (2016) assumes the system serves as a fallback level after that handover time, which Gasser et al. only assume for fully automated driving.

Full Automation

The SAE (2016) and the NHTSA (2013) assume for fully automated driving (or: NHTSA level 4, full self-driving automation) that all domains and driving situations can be handled by the automated vehicle. This is a stricter definition than in

Gasser et al. (2012). They assume that fully automated vehicles may also be limited to certain domains and situations. For instance, an automated vehicle that is only able to drive on highways could be fully automated towards Gasser et al. but not in the definition of the SAE level 5 of full automation or the NHTSA.

Autonomous Driving

Several groups use a wide definition of the term *autonomous driving* to refer to a similar range of functions as in automated driving. For instance, Maurer (2015, p. 2 ff.) uses "autonomous driving" deliberately to highlight the ethical and societal implications of automation. Reschka (2017, Ch. 1.3.2) and Reschka (2015) understand autonomous driving as a category beyond full automation as in Gasser et al. (2012) and possibly even beyond the SAE level 5 of full automation. True autonomy as in section 9.2 may be hard to achieve. Possible examples of true autonomy might be if a vehicle decides to leave its parking space for a trip to the next gas station because it strives to maintain its value for mobility for an anticipated upcoming long distance trip. As those levels of autonomy have little in common with the technical challenges being addressed in this thesis, the remainder of this thesis will use the term automated driving instead of autonomous driving. If not stated otherwise, the terminology of automation levels as in SAE (2016) will be used.

2.2 Systemic

General Definitions

As announced in the introduction, this thesis will take several factors into consideration for the issue of lane change behavior planning. Such a broader focus is often referred to as a *systemic* or *systems engineering* perspective. The Oxford Dictionaries (Oxford, 2017b) explains systemic as "relating to a system, especially as opposed to a particular part." The Collins Dictionary (Collins, 2017) defines systemic as "affecting the whole of something."

In Philosophy

Bunge (1979, p. 1) defines systemic as a "set of theories that focus on the structural characteristics of systems". To him, systemics has a "cognitive or theoretical rationale [...] to discover similarities of systems despite their specific differences" and a "practical motivation [...] in the need to cope with the huge and many-sided systems characteristics of industrial societies" (ibd.). Information about a system is specified by the "composition", the "environment", and the "structure of the system" as a "minimal model of a system". It may be supplemented by information about "the history of the system [...], and the laws of the system" (Bunge, 1979, p. 8).

This definition leads to distinguish between the system's components, its environment, and a system boundary to separate both (cf. "item definition" in the ISO 26262 standard (ISO, 2011, Part 3, p. 4)). In the context of lane change behavior planning, this system boundary may either be related to physical item boundaries (everything inside the automated vehicle vs. the outside) or may even entail several vehicles/entities forming a cooperative group for behavior evaluation (cf. Chapters 4 and 10).

Table 2.1: Levels of automation according to VDA (2015), SAE (2016), NHTSA (2013), and Gasser et al. (2012) with extensions as in Reschka (2017, p. 32) (environ. = environment, monit. = monitoring, w. = with)

Level (SAE)	**Designation**	**Actuator control**	**Decision making**	**Environ. perception**	**Fallback level**	**Driving situations**	**Monit. vehicle**	**Monit. freight**	**Supporting passengers**	**BASt / VDA**	**Level (NHTSA)**
Driver monitors environment, vehicle, freight, and supports passengers											
0	No automation	Driver	Driver	Driver	Driver	None	Driver	Driver	Driver	Driver only	0
1	Driver assistance	Driver & system	Driver	Driver & system	Driver	Some	Driver & system	Driver	Driver	Assisted	1
2	Partial automation	System	Driver & system	Driver & system	Driver	Some	Driver & system	Driver	Driver	Partially automated	2
System monitors environment; driver monitors vehicle, freight and supports passengers											
3	Conditional automation	System	System	System	Driver (w. handover time)	Some	Driver & system	Driver	Driver	Highly automated	3
System monitors environment and vehicle; driver monitors freight and supports passengers											
4	High automation	System	System	System	System	Some	System	Driver	Driver	Fully automated	3/4
System monitors environment, vehicle, and freight and supports passengers; operation without human driver in vehicle possible											
5	Full automation	System	System	System	System	All	System	System	System	VDA: driverless	3/4

2.3 Scene, Situation, Scenario, and Context [1]

Scene

Spatio-Temporal Relationships

Surprisingly inconsistent definitions exist for the relatively common term *scene*. Thomason & Gonzalez (1985, p. 26) propose a scene tree as a scene representation in which they decompose a scene into simpler elements and arrange those elements into a hierarchical structure. Maurer (2000, p. 63) defines a scene by the spatial-temporal arrangement of physical objects from an observer's point of view.[2] Geyer et al. (2014, p. 185) use an analogy to a theater to define: "A scene is defined by a scenery, dynamic elements and optional driving instructions. [...] A scene starts either with the end of the previous scene or – in case of the first scene – with a predefined starting scene."

Theater Metaphor

Temporal Duration vs. Snapshot

In the author's interpretation of Geyer et al. (2014) this means that a particular scene might persist for several seconds. For instance, a scene of the ego vehicle overtaking another vehicle might take several seconds before the scene changes into another one. This definition induces a technical challenge: It is hard or even impossible to fully determine when one of these several seconds spanning maneuvers will end. Thus, it is difficult to determine when the next scene should start, if it is not stipulated by a predefined update rate. Additionally, it is not clear in Geyer et al. (2014) if or how a starting scene differs from a regular, subsequent scene in terms of duration. Therefore, the authors suggest deviating from the definition of Geyer et al. (2014) in such a way that a scene is only considered as a *snapshot* of the environment's state and self-representation as described in Maurer (2000, p. 58 ff.). The *snapshot* concept does not contradict to include temporal aspects like the time since a previous event (e.g., overtaking a vehicle or being obstructed by a slow front vehicle).

Self-Representation

Geyer et al.'s (2014) definition suggests including "optional driving instructions" as part of the scene. Vice versa, according to Wershofen & Graefe (1996), the robot's goals should be part of the *situation*. Similarly, Haag (1998, p. 73) and Krüger (1992, p. 28) differentiate between a scene and a situation with the aspect of actions and possible action alternatives. Linked to this, the aspect of self-representation discussed by Maurer (2000, p. 58 ff.), Bergmiller (2015, p. 145 ff.), and Reschka et al. (2015) is not yet covered. For a systems perspective, it is relevant to consider not only the "environment" but also the "composition" and "structure" of the system itself (cf. section 2.2). Thus, a scene shall not only cover environment aspects, but also the aspect of self-representation. For automated driving, the author suggests making *goal-specific driving instructions* part of the situation, but add the idea of *self-representation* to the *scene* definition. The author suggests to understand Geyer's *driving instructions* just as information being part of the self-representation and not as *goals*. Thus, the author will use the term *scene* in the following way:

[1] This subchapter has been pre-published in Ulbrich et al. (2015g) and Ulbrich et al. (2015h). The coauthors provided an in-depth review and discussions. In particular, they added the aspect of incompleteness to the scene definition and a use-case definition. Moreover, they extended the scenario definition and backed it up by the citations.

[2] German: "räumlich-zeitliche Anordnung von physikalischen Objekten aus Sicht eines Betrachters".

> *A scene describes a snapshot of the environment including the scenery and dynamic elements, as well as all actors' and observers' self-representations, and the relationships among those entities. Only a scene representation in a simulated world can be all-encompassing (objective scene, ground truth). In the real world it is incomplete, incorrect, uncertain, and from one or several observers' points of view (subjective scene).*

Elements, Actors, and Observers

In this definition, an actor is an element of a scene acting on its own behalf. An observer[3] is a perceiving element within the scene or is observing the scene as a whole. An element might be an actor and an observer at the same time. Dynamic elements are elements that are moving or have the ability to move. The scenery subsumes all geo-spatially stationary elements (cf. section 8.1).

Subjects and Objects

Dickmanns (2007, p. 59) differentiates between objects and subjects. Subjects are defined as "bodily objects with the capability of measurement intake and control output depending on the measured data as well as on stored background knowledge." Whereas, objects are "having passive bodies and no capability of self-controlled acting" (Dickmanns, 2007, p. 446). In this thesis, objects will not be distinguished from subjects but rather grouped under the broader term of dynamic or static elements and attributed as actors and observers if needed. To the author, it is less relevant to differentiate between whether something has a "free will" (Dickmanns, 2007, p. 59), but rather whether something is dynamic or static. No matter if something moves because of its free will or because of the wind, it is relevant to avoid collisions. If a collision with a dynamic or static element cannot be avoided, moral rules may suggest rather to avoid collisions with subjects than with objects. Yet the free will is just one out of many dimensions as discussed in chapter 9. For instance, a collision with a bird as a subject may be tolerated in favor of not hitting an object like a bridge pole and risking the death of all passengers.

Subjectiveness and Implications

By being based either on observed information or a-priori-information that needs to be associated with observed information, a perceived scene will always be a subjective view of the world. Even if multiple observers share their information, it will not result in an objective representation of the world, but rather the view from multiple subjective observers. Thus, for a scene representation, an actor strives to achieve complete and certain information about the world, but in reality the scene will always be from one/several observers' points of view. However, in a simulated world a scene can be complete and uncertainty-free as from an omniscient observer's point of view.

Interface

A scene serves the basic purpose of an interface between environment- and self-perception modules on the one hand, and application- and mission-specific modules and tasks on the other hand. A sequence of scenes is a key part of a *scenario*.

Situation

Decision Relevance

While the usage of the term *scene* is inconsistent, the usage of the term *situation* is often even more undetermined. According to Wershofen & Graefe (1996, p. 3) cited by Maurer (2000, p. 95), a situation is the entirety of circumstances which are to

[3]This is not an observer as in the sense of control engineering.

be considered by a robot for its selection of an appropriate behavior pattern in a particular moment.

In psychology, Wirtz (2015c) defines a situation as the entirety of circumstances which results in a certain behavior of a human. Wirtz uses the term situation for a person plus its psychological setting.

External and Internal Situation

Reichardt (1996, p. 35) cited by Maurer (2000, p. 95) defines a situation as the union of subsets of the internal and external situation. The internal situation consists of a subset describing the (automated) vehicle's state and its user input. The external situation consists of the environment information describing the street, obstacles and traffic signs. He limits his situation definition to the "world of discourse[4]" where the automated vehicle is used. He clarifies that this is just a subset of the real world.

Aspect of (Possible) Actions

According to Haag (1998, p. 73) cited by Pellkofer (2003, p. 4), the difference between a *scene* and a *situation* is the *aspect of (possible) actions*.[5] Krüger (1992, p. 28) also cited by Pellkofer (2003, p. 4) defines a situation as an extended (system) state, in which an element is not only seen as a physical object, but also its *actions* and *action alternatives* are considered to estimate the temporal development of a situation.

Pellkofer (2003, p. 4) defines a *situation* as the sum of all behavior decision relevant aspects. Relevant for the behavior decision making are the current scene, the intentions and actions of all subjects in a scene (including the ego vehicle), and the abilities of the ego vehicle, which represent the decision alternatives. In contrast, the author does not consider *abilities* as *decision alternatives*, but as *input* to the decision process to derive decision alternatives.

Mock-Hecker (1994, p. 4) cited by Maurer (2000, p. 95) considers the traffic situation to be an extract of the traffic (world) at a *certain point of time*. It entails the actions and plans of traffic participants. To him, a situation not only represents the current state but also its probable future development.

Snapshot

Once more, an important aspect of the technical usage of a *situation* is its applicability. As for the scene, the author suggests considering a *situation*, similar to Mock-Hecker (1994, p. 4), as a snapshot of the entirety of circumstances, which are to be considered by a robot (actor) for the selection of an appropriate behavior pattern in a particular *moment*. Revisiting the overtaking example from the scene definition, a situation would not last for the several seconds that an overtaking scenario might take, but would be a *snapshot*. Again, such a definition avoids the technical challenge of determining what kind of situation it currently is and how long it will last before the world changes into another situation.

Situation Aspects

Another challenge for the definition of the term *situation* arises from the system architecture elements called "situation assessment" or "situation analysis" as a submodule of the guidance block (cf. Figure 3.1 in Chapter 3). A situation assessment uses a situation as input and interprets the situation or particular aspects of it. Thus, the results may be considered as an augmentation of the prior situation, which pro-

[4]German: "Diskurswelt".
[5]German: "Handlungsaspekt".

vided further details regarding certain aspects. Reichel et al. (2010), Reichel (2013, p. 63 ff.), Siedersberger (2003, p. 140), and Pellkofer & Dickmanns (2002, p. 47 f.) coined the term "situation aspects" for these.

Eco (1972, p. 65) cited by Maurer (2000, p. 95) considered the transition from a signal to a meaning[6] as the central signification process conducted by humans. According to Maurer (2000, p. 95), a situation assessment could be considered as such a signification process in a technical system.

Signal to Meaning

Geyer et al. (2014) clarify that a "situation is defined by the set of criteria, that need to be true to conduct an associated action." As for their scene definition, the end of a situation is defined by a change of one criterion, that describes the situation. The author agrees with Geyer et al. (2014) that, "depending on the action, the same scene can evolve into different situations." In the illustration of the proposed ontology in Geyer et al. (2014, p. 185), the situation seems to fully entail a scene. To the author, the instructions in Geyer's scene should be part of the situation and not the scene. Moreover, the situation should result from information *selection and augmentation* of the scene information based on the mission-specific or permanent (cf. Wershofen & Graefe, 1996) goals and values of the automated vehicle.

Obtaining a Situation from a Scene

Angenendt (1987, p. VIII) assumes a situation is more than just a snapshot of the traffic scene[7] with the infrastructure and environment representing measures. Above this, a situation contains information regarding the *behavior* of traffic participants and the resulting informal rules of conduct. He uses the concept of a *behavior setting* to entail the informal rules followed by traffic participants (Angenendt, 1987, p. 23). The author agrees that the "behavior setting" is an integral part of the situation. It is subsumed under goal- and value-related information.

Behavior Setting

Von Benda (1985, p. 1) defines a traffic situation as a limited extract of the entire traffic scene. A vehicle's driver sees such an extract from his or her limited point of view. A traffic situation from a driver's point of view is to her the environment of the human machine system. She assumes that a situation pertains for a certain amount of time until a new situation starts with the interaction with a new event.

Extract from a Point of View

Dickmanns (2007, p. 448) defines a situation as "the collection of environmental and all other facts that have influence on making proper (if possible 'optimal') behavior decisions in the mission context. This also includes the state within a maneuver being performed [...] and all safety aspects." For the author's situation definition, the author agrees with Dickmanns' driving function relevance criteria for situation information.

Schmidt et al. (2014) differentiate between a *true world model*, a *true situation* for an individual observer and a *subjective situation* from an individual observer's point of view. While agreeing that a true situation may exist in a perfect simulated world, a real situation representation in a technical system will always be not all-encompassing, uncertain, and from a subjective point of view (cf. scene definition).

Subjectiveness

[6] German: "Übergang vom Signal zum Sinn".
[7] German: "Verkehrsgeschehen".

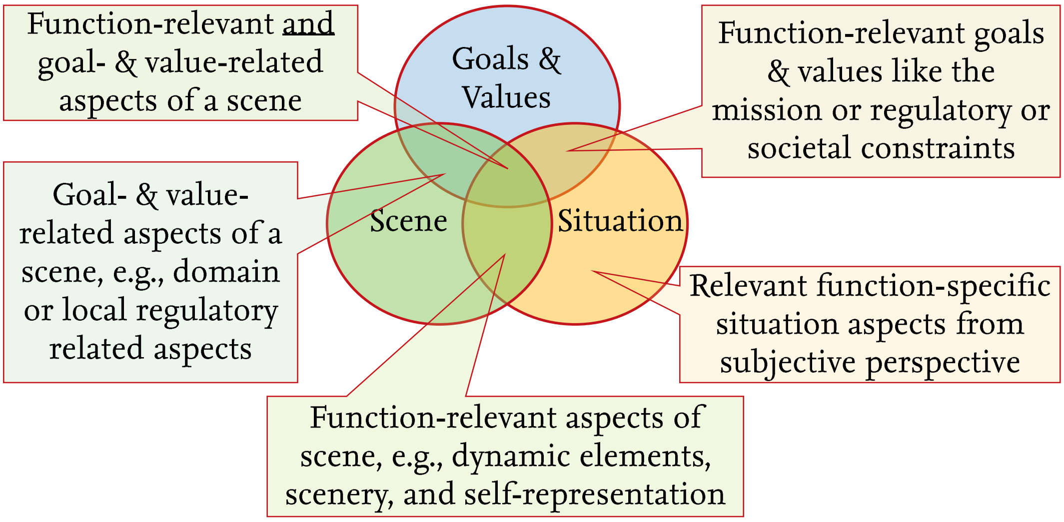


Figure 2.1: Venn diagram of scene, situation, and an element's goals and values

While acknowledging that it is hard to find a general definition of the term *situation*, the author suggests the following definition:

> *A situation is the entirety of circumstances, which are to be considered for the selection of an appropriate behavior pattern at a particular point of time.*[8] *It entails all relevant conditions, options, and determinants for behavior.*[9] *A situation is derived from the scene by an information selection and augmentation process based on transient (e.g., mission-specific) as well as permanent goals and values. Hence, a situation is always subjective by representing an element's point of view.*

A situation consists of several situation aspects to be interpreted or comprehended by situation assessment modules. A situation is the input and output of such modules at once.

Relationship between Scene, Situation, and Goals & Values

According to the author's definition of a situation, it can be fully derived from a scene and the system's goals and values, as illustrated by the Venn diagram in Figure 2.1. There is a wide overlap between a scene and a situation to include, e.g., all relevant parts of the scenery, all relevant dynamic elements, and all relevant aspects of the self-representation. This *information selection* helps to simplify the situation representation and by this the driving function development and computational complexity. Moreover, the situation is implicitly or explicitly *augmented*, e.g., by goals and values. For instance, by explicitly labeling the usefulness of roads or lanes to reach the mission goal or implicitly by characterizing a child playing on the side to be more relevant than a plastic bag that is flying around. The remaining part of the situation, not overlapping with the scene or the goals and values, represents situation aspects that are evaluated and populated with information by situation assessment modules.

[8] Cf. Wershofen & Graefe (1996).
[9] Cf. Meyer (1977). Determinants as in determining factors.

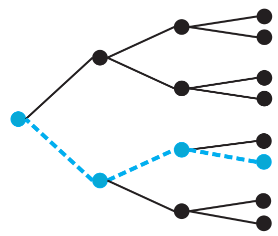


Figure 2.2: A scenario (dashed blue line) as a temporal sequence of actions/events (edges) and scenes (nodes)

Scenario
The term "scenario" is often found in the context of simulation and testing, and in the functional description of driver assistance systems.

Scenarios as a Term in many Disciplines

According to Jarke et al. (1998, p. 155) there are "three major disciplines that use scenarios - strategic management, human-computer interaction, and software and systems engineering - to deal with [the] description of current and future realities." Go & Carroll (2004) remark that the usage of scenarios in any field is quite different, but the elements of a scenario are similar. According to Go & Carroll (2004, p. 46), "a scenario is a description that contains (1) actors, (2) background information on the actors and assumptions about their environment, (3) goals or objectives, and (4) sequences of actions and events."

Sequential Character

The Oxford Dictionary (Oxford, 2015d) defines a scenario as a "postulated sequence or development of events" or the "written outline of a film, novel, or stage work giving details of the plot and individual scenes."

Geyer et al. (2014) define that "a scenario includes at least one situation within a scene including the scenery and dynamic elements. However, [a] scenario further includes the ongoing activity of one or both actors. According to the movie and theater metaphor previously introduced, the term scenario can be understood as some kind of storyline – including the expected action of the driver – but does not specify every action in detail." To the author, Geyer et al. should consider *multiple* actors, instead of *both*, for a general definition.

According to Figure 2.3, a scenario contains scenes, actions & events, and goals & values. The author suggests the following definition:

> *A scenario describes the temporal development between several scenes in a sequence of scenes. Every scenario starts with an initial scene. Actions & events as well as goals & values may be specified to characterize this temporal development in a scenario. Other than a scene, a scenario spans a certain amount of time.*

Spanning of Time

Scenes in a scenario are *linked* by actions and events. According to Figure 2.2, a scenario is a single path of a *temporal sequence* of actions & events (edges) and scenes (nodes) out of the tree representing the entirety of all possible future scenarios for a given initial scene. Unlike scenes, a scenario spans a certain *amount of time*. A scenario needs to include *at least one* (initial) scene and actions & events to fully specify a path in Figure 2.2. However, a scenario may also be specified by a complete set of scenes, while the actions and events just cover the elapse of a specified time.

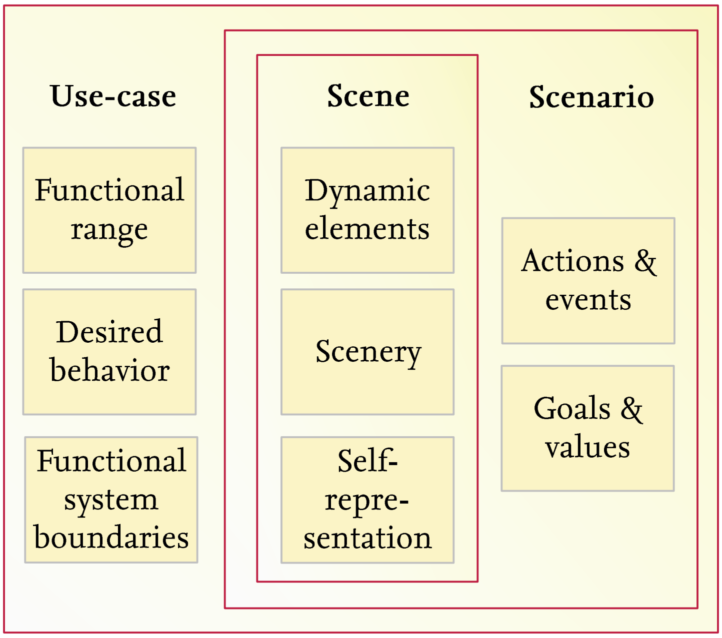


Figure 2.3: Scene, scenario, and use-case

Actions, Events, and Key Scenes

Using the theater metaphor, a scenario is typically described by *several* scenes with prescribed actions & events in between. In the real world, any actions & events are to some degree uncertain. Hence, human actors may even slightly adopt their behavior to achieve a prescribed key or closing scene in a theater play. Likewise, a driving scenario may contain certain key scenes, e.g., a narrowly defined crossing sequence at an intersection. In the extreme, a scenario is described by a storyboard describing every minimal detail as in a cartoon movie. Vice versa, a scenario may also only specify that all actors shall start working towards their goals while following a certain set of prescribed actions & events without specifying any future scenes. However, possibly allowing uncertainty in the behaviors may result in an entirely open outcome of such a scenario after the elapse of a certain amount of time. If no uncertainty is allowed, both forms of description will be a dual way to specify exactly the *same* scenario.

Specifying Situations only

Depending on what a scenario is used for, it may also be sufficient to specify *only* situations instead of entire scenes plus goals and values. This may be true for a test setup only designed to test, e.g., a situation assessment as in the situation-based open-loop test described in Chapter 13.

Test-Cases and Use-Cases

For simulation and testing of an automated vehicle or its modules, test-cases may be specified. Each of them entails a scenario and pass-fail criteria to evaluate it. Furthermore, the functional description of the system (use-case) needs to be defined in the early phases of the system design according to the V-model, e.g., in the ISO 26262 standard development process (ISO, 2011, Part 3). A use-case entails a description of the functional range and the desired behavior, the specification of system boundaries, and the definition of one or several usage scenarios. While these scenario descriptions might be rough and incomplete in the first phase, they may be detailed to achieve fully testable test-cases in the development process.

Context
The term “context” or the process of “context modeling” or “world modeling” is often used to describe activities of aggregating environment information into an abstract context model. The Merriam-Webster Dictionary explains that a context is described by “the interrelated conditions in which something exists or occurs” (Merriam-Webster, 2015a). In language science it is “the parts of a discourse that surround a word or passage and can throw light on its meaning” (Merriam-Webster, 2015a). Wirtz (2015a) defines a context as the surrounding conditions, which contribute to the meaning of an event or information. Dey (2001) states “context is any information that can be used to characterize the situation of an entity.” To him, “an entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.”

In this thesis, the term context is used as a general phrase to describe *the part of discourse that surrounds and represents an element*. This entails the scene with all its components. Above this, the context is assumed to span at least the lifetime of the element. Thus, it entails many scenarios. Due to this wide definition, a full context may never be represented in its entirety. Broad Definition

How can the term *context* be distinguished from the term *domain*? According to Pellkofer (2003, p. 5), a domain is the class of environments, in which whole groups of actions are allowed, necessary, or forbidden. To the author, the domain is part of the context. It summarizes aspects that distinguish, e.g., an urban domain from a highway domain. Yet, to the author a domain is rather a set of aspects that hold true for a certain amount of time but do not necessarily need to span the lifetime of an element. Domain

Further discussions can be found in Ulbrich et al. (2014), Nothdurft (2014, p. 31), Brown (1995), Dey & Abowd (2000), and Strang & Linnhoff-Popien (2004).

The term context remains vague to some extent. As a context – by this definition – subsumes pretty much everything, it is questionable what use it actually provides. It will be used to refer to the general process of *obtaining information about the context* and to generate *scenes* as snapshots from it. Challenge

2.4 Maneuver, Behavior, Intentions, Skills, and Abilities

Maneuver
The word maneuver originates from the Latin terms *manus* (hand) and *operari* (to work) and was used in the mid-18th century as a noun in the sense of a “tactical movement” (Oxford, 2015c). The Merriam-Webster Dictionary defines a maneuver as “an action taken to gain a tactical end” (Merriam-Webster, 2015c). The Oxford Dictionary explains it is “a movement or series of moves requiring skill and care” or “a carefully planned scheme or action [...]” (Oxford, 2015c). Origin

Dickmanns (2007, Chapter 2.1.4.2) defines maneuvers “as typical time histories of control outputs for achieving desired transitions from one regime of steady behavior to another”, which may “last up to several minutes.” In a similar sense, the term maneuver is used in this thesis for a *sequence of (tactical) actions*. Typical maneuvers In Automated Driving in the field of automated driving are for instance overtaking another vehicle or performing a lane change.

Mission Element

Maurer (2000, p. 44 ff.), Gregor (2002, p. 43 ff.), and Siedersberger (2003, p. 42, p. 142 ff.) elaborate on the idea of *mission elements* to transform goals- and value-based missions into smaller, executable chunks of behavior. Dickmanns (2007, p. 448) defines that they are “those parts of an entire mission that can be performed with the same subset of behavioral capabilities and parameters”. In fact, to the author every mission element entails one or more, trivial or abstract maneuvers. Maneuvers are “triggered by higher level decisions for implementing strategic mission elements [...] or due to the actual situation encountered” (Dickmanns, 2007, p. 448).

Behavior

Origin

The term *behavior* is part of everyday common English. The Oxford Dictionary defines behavior as “the way in which an animal or person acts in response to a particular situation or stimulus” or “the way in which [...] a machine works or functions” (Oxford, 2015a). In psychology, Kaiser (2014, p. 1623) explains that behavior subsumes “every form of motoric activity” and is, by its “efferent-innervated muscle activity”, accessible to objective examination. Gindele et al. (2010) define that “behaviors are symbolic representations of context dependent motion primitives that a vehicle is able to conduct.”

In this Thesis

In this thesis, the term is used to *subsume any kind of actions or activities that result in observable outcomes.* This entails behavior on a strategic, tactical, and stabilization level of any element.

Intention

Definition

According to the Merriam-Webster Dictionary, an intention is “a determination to act in a certain way” (Merriam-Webster, 2015b). In psychology, Puca (2015) defines an intention as a determination or resolution to pursue a certain action or to reach a certain goal.

Characteristics

According to Wooldridge (1999), the typical characteristics of an intention are that they “drive a means-ends reasoning”, “constrain future deliberation”, “persist” for a longer amount of time, and “influence beliefs upon which future practical reasoning is based.”

Mock-Hecker (1994, p. 5) notes that critical situations may result from incorrect action execution on an operational level, but more importantly also from conflicts between tactical intentions.

Abilities and Skills [10]

Abilities

According to Häcker (2015), an ability is defined as the entirety of conditions which are necessary to deliver a performance. Carroll (1993, p. 4) states an ability expresses the potential of achieving something if conditions are favorable. He defines ability as “the possible variations over individuals in the liminal levels of task difficulty [...] at which, on any given occasion in which all conditions appear favorable, individuals

[10] An early version of this section has been pre-published by the author in Reschka et al. (2015). Several of the citations have initially been found by Bergmiller (2015). The author contributed the link towards Häcker (2015) and Heuer (2015) and the subcategories of cognitive abilities based hereon. Moreover, the illustrative example has been added in this thesis.

perform successfully on a defined class of tasks." Hirtz (2003a, p. 188) defines abilities as relatively consolidated, more or less generalized, individual qualifications for performing certain tasks or actions in the context of sport sciences.

Skills

Vice versa, skills represent a, by repetitions and training, more or less automated component of a task or action, which is based on an underlying ability (Hirtz, 2003b, p. 196). According to Heuer (2015), a skill describes a task-related activity and includes a performance level (skill level). Further, (human) skills can be separated into (senso-) motoric, cognitive, cognitive motoric, social, linguistic, and perceptive skills.

In Automated Driving

Maurer (2000, p. 101 ff.), Pellkofer (2003, p. 62 ff.), and Siedersberger (2003, p. 73 ff.) elaborate on the concept of an automated vehicle's skills and abilities for self-representation, performance monitoring, and system skill hierarchies. In relation to vehicular systems Heuer's (2015) categories "motoric", "cognitive", "perceptive", and – according to cooperative systems – "social" also seem to be applicable. Bergmiller (2015, p. 145 ff.) mainly refers to motor abilities, as his work focuses on representing observable actions carried out by the vehicle but also outlines the relevance of cognitive abilities. He takes Carroll (1993, p. 3 ff.) into account to define cognitive abilities as a counterpart to motor abilities. The authors in Reschka et al. (2015) propose refining the understanding of cognitive abilities and point out that automated vehicles have, for now, no learning ability as humans do. The subcategories of cognitive abilities for automated vehicles can be defined as perceptive (for environment and self-perception), planning (for decision making and trajectory planning), and social (for upcoming cooperative systems).

At this point, one may feel unsatisfied that there is still no single, concise definition for "skills" and "abilities". Yet, Bergholz (2003, pp. 43–104) summarizes research on these two terms in the disciplines of sport science, psychology, and educational science on over sixty pages only to conclude that there is simply no consistent use of the two terms.

Example

Instead of trying to come up with *the* definition, a simple example shall be provided: A sensor of an automated vehicle may generally have the *ability* to emit signals and detect reflected signals from trucks 200 m ahead according to its maximum performance specifications, but the current *skills* may be degraded to a lower level of perceiving objects 70 m ahead due to dirt, fog, or general sensor wear and tear.[11]

2.5 Types of Uncertainties

Uncertainty and Risk in Decision Making

Addressing uncertainty is central for behavior planning for automated vehicles. Yet, the term is to some extent vague. In economics (Knight, 1964, p. 19 f.) and in cognitive psychology (Gigerenzer, 2014) it is common to differentiate between deci-

[11] Opposite to this example, Reschka (2017, p. 65) redefines the terms skill and ability for technical systems. For him, a sensor may generally have the *skill* to emit signals and detect reflected signals of objects 200 m ahead, but its current *abilities* are degraded to a lower level. This is exactly vice versa to the usage in Reschka et al. (2015), Bergmiller (2015), Hirtz (2003b), and (Hirtz, 2003a). Given the ambiguity in the literature, this is not wrong. Yet, the author of this thesis sticks to the prior more common usage.

sion making under *risk*[12] and decision making under *uncertainty*. Decision making under risk means all options, consequences and probabilities are known or can be estimated. Typical approaches are known from game theory. If not all of these conditions hold true, this is called decision making under uncertainty.

Perception Uncertainty

Different aspects contribute to uncertainty. Mählisch (2009, p. 5) differentiates in environment perception between *existence* uncertainty, *measurement* uncertainty, and *association* uncertainty. The first accounts for the uncertainty in object detection by sensor systems, because not all detected measurements are necessarily caused by truly existing elements. Vice versa, elements may exist without being supported by measurement data. The second kind of uncertainty reflects the errors in measuring physical state variables and parameters itself, like distances, velocities, etc. They result from measurement errors of the sensor systems itself. The third kind of uncertainty reflects the ambiguity of how to associate measurement data to elements in an environment model.

Execution Uncertainty

Further more, there is uncertainty induced by the behavior execution and planning itself. *Actuation* involves uncertainty from wear-and-tear, control noise, and mechanical failure (Thrun et al., 2005, p. 4). Any representation in *internal models* is approximate and induces model errors. Likewise, behavior decision making for real time systems requires *algorithmic approximations* in complex domains. These approximations result in errors to be reflected in increased uncertainty.

Prediction Uncertainty

Another type of uncertainty they point out is resulting from the inherently *unpredictable future* of an environment (Thrun et al., 2005, p. 4). Due to only partial observability and interactions of elements, predictions in a non-deterministic world are inherently wrong. Not only propagates the perception uncertainty into the future but even the uncertainty about an element's *goals, values, and intentions* renders longer predictions uncertain. Even if intentions were known, disturbances to the traffic system would render predictions to be inherently wrong. For instance a moose running on a street may cause a prediction uncertainty even in a traffic system with ubiquitous Vehicle-To-X communication and perfect knowledge about the intentions of the other traffic participants.[13]

2.6 Conclusions

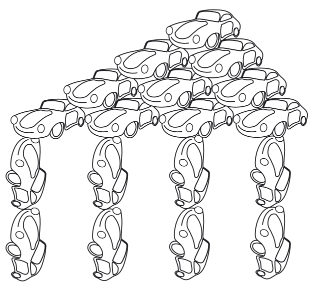
This chapter established an understanding of what automated and assisted driving is and pinpointed the challenge of system monitoring and the role of a human system operator.

The terms scene, situation, scenario, and context were defined to clarify the interfaces and necessary inputs for tactical behavior planning and the testing of such modules. While it was possible to find a consistent definition of the first three terms, the definition of the term context remains vague to some extent. As a context is defined in such a broad manner, it is questionable what use the term actually has.

[12] In this domain, the term "risk" is used differently than in the ISO 26262 (ISO, 2011, Part I, p. 13).
[13] Special thanks to Prof. Markus Maurer for pointing out this illustrative example.

The terms maneuver, intention, and behavior are central to address the core of this thesis for tactical behavior planning. The terms seem less controversial than the terms in section 2.3, nevertheless it was necessary to define them clearly and more specifically than in regular common English. The terms skills and abilities are discussed as a foundation for skill and ability monitoring in situation assessment. Last of all, the term uncertainty is substantiated.

3 Behavior Planning in a Functional System Architecture for Automated Vehicles [1]



Functional system architectures describe the composition and structure of a system (cf. section 2.2). They are core to structure the technical development of software for automated vehicles. When software is developed for an automated vehicle, it is often a bottom-up process. If existing building blocks are just *hacked* together in some way this will lead to complex system designs. Yet, having a well structured functional system architecture is key. It has central impact on the system design and technical software development for an automated vehicle — often for several years. This chapter presents an overall functional system architecture for an automated vehicle. Implementation independent modules are grouped such that there are clean interfaces among these modules. This functional system architecture differs from others by strictly using hierarchy and functional separation. It underwent several iterations. Earlier versions of this architecture have been published by the team at TU Braunschweig.[2] It has strongly been influenced by the functional system architecture developed by Dickmanns' group.[3] The following architecture discussions will be based on the last architecture revision in Matthaei & Maurer (2015) and some refinements in Matthaei (2015, p. 37 ff.). Apart from that, several other enhancements have been made to incorporate more recent research such as the definition of interfaces, e.g., the definition of a *scene* or *situation* in section 2.3, or functional safety considerations resulting from the application of the architecture in the aFAS project (Stolte et al., 2015).

Structure

This chapter is organized as follows. First, different approaches for structuring driving tasks and their processing levels are introduced and compared with other

[1] This chapter has been pre-published by the author in Ulbrich et al. (2017b). The coauthors provided an in-depth review and contributed to several improvements such as the reorganization of the localization and map provisioning column, the introduction of the execution monitoring, changes in the perception interfaces of the column, and the reorganization of navigation and guidance. In particular, the coauthors helped to improve the argumentation of how the Vehicle-To-X communication is integrated towards the map provisioning and perception. They clarified the role of predictions within the architecture and straightened out the role of the blocks within the context modeling. Regarding the guidance and stabilization block, the coauthors clarified the interfaces and streamlined the text. With the coauthors' help it was possible to identify and clearly write down the enhancements over previous versions of the architecture and to challenge each and every proposed modification until profound arguments were identified. Apart from the coauthors, in particular Fabian Schuldt contributed in restructuring the localization and map provisioning column. The whole team at the Institute of Control Engineering took part in numerous discussions to reach the status quo.

[2] Matthaei (2015), Matthaei & Maurer (2015), Nothdurft et al. (2011), Reschka et al. (2011), Rieken et al. (2015), Saust et al. (2010), Ulbrich et al. (2014), Wille (2012), and Wille et al. (2010).

[3] Dickmanns et al. (1994), Dickmanns (2007), Hock & Dickmanns (1992), and Maurer (2000).

approaches in the literature. Then the functional system architecture is presented by outlining its main columns and clarifying its interfaces and comprised activities. For each aspect the modifications to the state of the art are presented and provided with a root cause for these. In the end, open issues are highlighted and a conclusion is drawn.

3.1 Background

Requirements

The ISO 26262 standard proposes a functional system architecture as a part of the system design and defines "modularity", "adequate level of granularity", and "simplicity" as requirements to the architectural design (ISO, 2011, part 4, p. 13). As a property of a modular system design the ISO 26262 proposes "hierarchical design", "precisely defined interfaces", "maintainability", "testability", and "avoidance of unnecessary complexity" (ibid.).

Table 3.1: Examples for driving tasks and their processing levels based on Hale et al. (1990, p. 1383) depicted as in Muigg (2009, p. 8)

		Processing level		
		Skill-based	**Rule-based**	**Knowledge-based**
Driving task	**Navigation**	Daily commute	Choice between familiar routes	Navigating in strange town
	Guidance	Negotiating familiar junctions	Passing other car	Controlling a skid on icy roads
	Stabilization	Road following around corners	Driving an unfamiliar car	Learner on first lesson

Rasmussen

To provide a structure for human behavior, Rasmussen (1983) distinguishes "skill-based behavior", "rule-based behavior", and "knowledge-based behavior" as three levels of performance of skilled human operators. On the lowest, skill-based level, reactive, sensory-motor activities take place without conscious control. On a rule-based level, decisions are taken based on a previously stored set of rules. If a situation is not familiar and there is no stored rule for it, knowledge-based behavior may be applied. Here a new strategy for goal archival is developed from existing knowledge.

Donges

Donges distinguishes "navigation", "guidance", and "stabilization" as three hierarchical levels of driving tasks in his publication from 1982 cited in Donges (1999). Similar to Riemersma (1979) and Michon (1985, p. 489) citing his inaugural lecture from 1971, one elementary (operational) layer is used for course keeping and speed control, a second (tactical) layer is for any behavior planning and a third one is for strategic planning.

Hale et al. (1990, p. 1383) suggest that the three levels of driving tasks and the three levels of Rasmussen are rather orthogonal to each other. While Donges' driving tasks address *what* task is to be solved, Rasmussen addresses *how* it is solved. Table

3.1 illustrates the relationship between both. Most of the time stabilization tasks are handled on a skill-based level. However, if for example a car is unknown to the driver, he or she may not have these subconscious skills to address the task. Yet, there are learned rules that may be used. If even these rules needed to be formed from knowledge because it is the first time a fresh learner drives a car, it would be knowledge-based behavior. On a tactical level, passing another car typically involves situation assessment and a certain amount of stored rules and experience, how much of a gap is necessary to overtake, etc. Humans address this typically by rule-based behavior. Last of all, navigation may display the widest variety of processing levels in everyday driving. While blindly commuting the same street every day without looking for, e.g., changed traffic signs may almost be skill-based behavior, it becomes rule-based if it involves active tactical decisions between several route options and turns into knowledge-based behavior if a driver navigates in an unknown city for the first time.

Driving Tasks and Processing Levels

Transferring the concept of driving tasks and processing levels from human drivers to a technical system provides a starting point for a technical architecture. In fact, this provides a hierarchical abstraction of driving tasks as in Maurer (2000) and Matthaei et al. (2015). Another distinction of tasks for automated driving may be derived from different processing steps of perceiving and acting.

Foundations

Extending work by Zapp (1988), who described a functional control-cycle for automated vehicles, Hock & Dickmanns (1992) showed an inverted "U" shaped signal flow from sensors to actuators with a hierarchical separation of processing levels for driving tasks. Further specification and an exhaustive system description can be found in Dickmanns et al. (1994). For instance in Dickmanns & Graefe (1988, p. 239) and more clearly in Dickmanns (2007, p. 441), Dickmanns presents the concept of separating "recognition" from "behavior (execution)" as well as the idea of aggregating features into abstract symbolic representations. Maurer (2000) and Dickmanns (2007, p. 185) highlight multiple feedback loops at different hierarchical levels constituting the signal flow in the architecture nowadays. In Dickmanns & Graefe (1988), Dickmanns & Müller (1996, p. 595), and Dickmanns (2007, p. 387), Dickmanns also illustrates the usage of a dynamic knowledge base and background knowledge that is now named as context modeling (cf. section 2.3) in this architecture. Additionally, Maurer (2000, p. 40 ff.), Siedersberger (2003, p.73 ff.), Pellkofer (2003, p. 64), and Dickmanns (2007, p. 442) transfer a system architecture into different capabilities for an automated vehicle. Dickmanns (2007, p. 442) separates between "scene understanding", "planning", and "gaze and locomotion control". As in Maurer (2000), the situation assessment is rather part of the perception. In Dickmanns et al. (1994), situation assessment stretches into both worlds: the planning column as well as the perception column. In this thesis, the goals- and value independent scene/context modeling is considered to be part of the perception column. The goals- and value specific situation extraction and situation assessment is considered to be part of the planning and control column. The idea of "situation aspects" as a result of a "situation assessment" is presented in Pellkofer (2003, p. 51 ff.) for automated driving.

Literature Review

Matthaei (2015, p. 25 ff.) provides a more comprehensive literature review on different forms of a functional system architectures that have been used by different

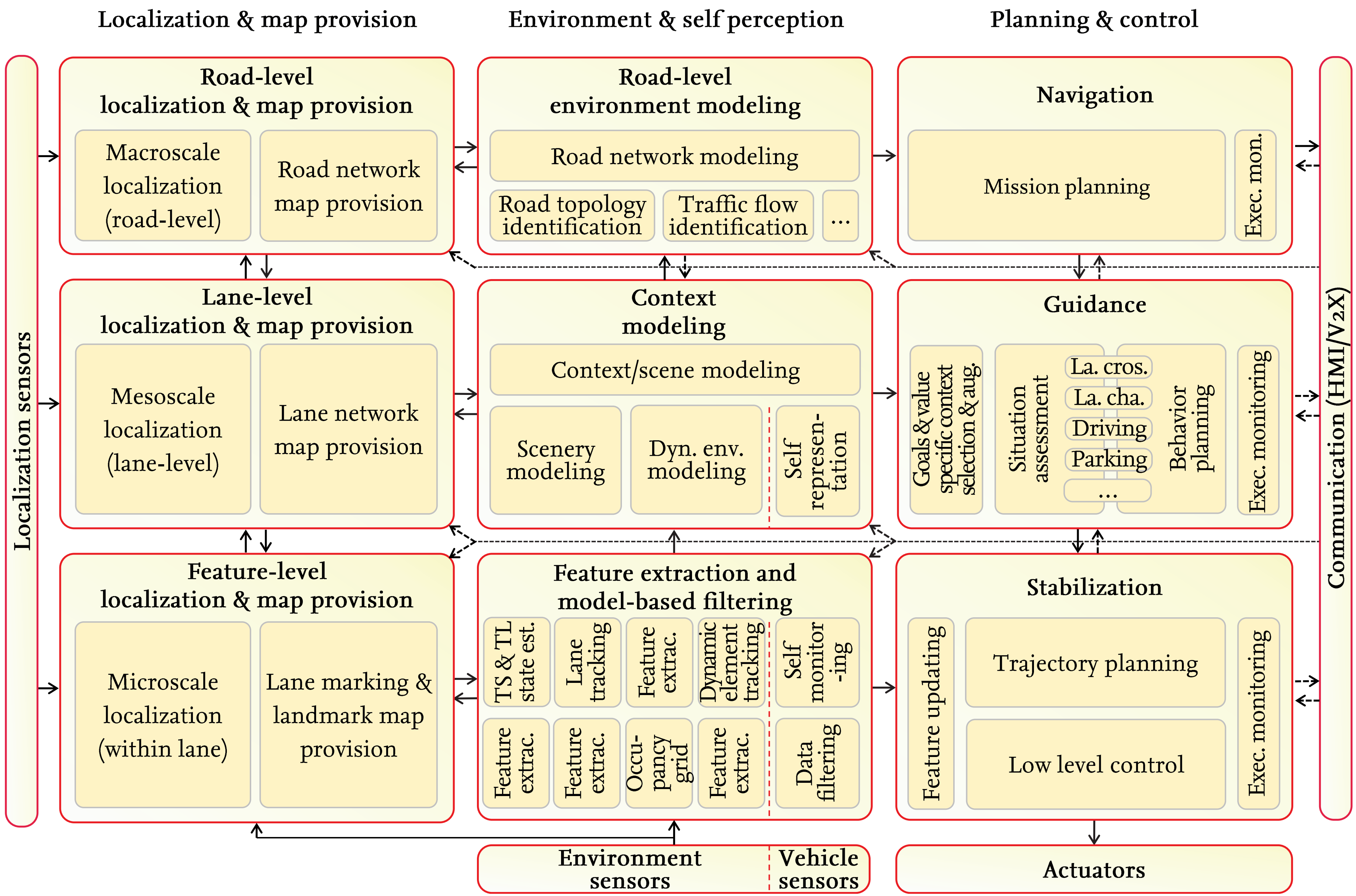


Figure 3.1: Functional system architecture of an automated vehicle. Blocks represent modules for activities; arrows show information flows (TS & TL state est. = traffic sign & traffic light (recognition and) state estimation, feature extrac. = feature extraction, dyn. env. = dynamic environment, aug. = augmentation, la. cros. = lane crossing handling, la. cha. = lane change handling, exec. mon.= execution monitoring, HMI = human machine interface, V2X = Vehicle-To-X)

teams in automated driving as well as in robotics. For a broad literature review the reader is referred to his dissertation. Yet, the author would like to relate the chapter to some recent publications on functional system architectures for automated vehicles.

Recent Publications

Tas et al. (2016) compare the functional system architectures of several automated vehicles. They summarize the advantages and disadvantages of a distributed, modular architecture such as this. They highlight the importance of fault detection, diagnosis, and self monitoring for system robustness in automated driving. Further, they unify the visual representation of three other vehicle architectures (Jo et al., 2015; Kunz et al., 2015; Wei et al., 2014) into a common visualization scheme. Here, Jo et al. (2015), Tas et al. (2016), and Wei et al. (2014) use a hierarchical structuring of driving tasks as in Donges (1999) for "mission planning", "behavior and motion planning", and "vehicle control and actuation". Perception, localization and "vehicle state estimation" (cf. "self monitoring") are not hierarchically structured in Tas et al. (2016), Jo et al. (2015), or Kunz et al. (2015). A "scene understanding" or "environment model" seems to be understood similarly to "context/scene modeling" as a central point for information aggregation. Tas et al. (2016) suggest to consider Vehicle-To-X (V2X) communication as an "array of redundant sensors". If such an approach is chosen, it is of particular importance to keep in mind that V2X information can be uncertain, incorrect, and even intentionally misleading. Further V2X communication might provide information on different levels of abstraction. Hence, the author treats information from V2X differently than information from onboard sensors (cf. section 3.2.3).

Behere & Törngren (2015) identify core components in a functional system architecture and group them under "perception", "decision and control", and "vehicle platform manipulation". Their components resemble mostly the components here. Yet, not all of the components here are part of their architecture. Additional components they identified are "energy management" for "battery management" and "regenerative braking" and "reactive control" for reflex responses to unexpected stimuli as in automated emergency braking. Here, energy management is considered as part of the vehicle. Reactive control is indeed considered. It is part of the stabilization module and its low-latency data link explained in section 3.2.2. Like Tas et al. (2016), Behere & Törngren (2015) consider Vehicle-To-X communication similar to a sensor/actor. Finally, Behere & Törngren (2015) identified the otherwise often neglected aspect of "diagnosis and fault management". The author agrees to its importance. In this architecture it is part of "self perception" (cf. section 3.2.1) and "execution monitoring" (cf. section 3.2.2).

Felix Lotz (2017, p. 128 ff.) develops a reference architecture for automated and assisted vehicles. For planning and control he follows a hierarchical structure as in Donges (1999). Similar to this publication, he proposes a human machine interface directly linked towards the planning and control column. Also in agreement with this publication, he proposes perception and context modeling aspects to form a column left to the planning and control column. He coins this column a "world model" and considers mapping and map provision as part of this column. Moreover, similar to this architecture, he proposes a "scene" as an interface between perception and planning and control. Other than in this thesis, he does not use a situation representation. He proposes a "mode planner" to command different behaviors to a behavior planner. In this thesis, this mode planner is not distinguished from the behavior planning module. Last of all, he considers trajectory planning not as part of the stabilization layer but rather of the guidance layer. Apart form the system architecture itself, he develops as set of requirements for an architecture, provides a detailed interface description, and develops a scenario catalog for automated vehicles.

Link towards Own Research Projects and Publications

The functional system architecture presented here has been inspired by and applied to the Stadtpilot project for automated driving in urban environments and the aFAS project for an unmanned protective vehicle for highway hard shoulder road works. Due to its generality, several aspects of the functional system architecture similarly apply to the *Audi A7 piloted driving concept* vehicle.[4] Yet, it has initially been developed as part of the Stadtpilot project. Hence, it is not just a top-down concept from a sketch board but has actually been proven to work in real world automated driving. It underwent several iterations. The foundations have been laid in Reschka et al. (2011), Saust et al. (2010), Wille (2012), and Wille et al. (2010), the concept for context modeling has been developed in Matthaei & Maurer (2015), Nothdurft et al. (2011), Ulbrich et al. (2014), and Ulbrich et al. (2015g), environment perception has been refined in Matthaei (2015), Matthaei & Maurer (2015), and Rieken et al. (2015), self representation has been addressed by Reschka (2017), Reschka et al. (2015), and Stolte et al. (2015), and localization and map provision has been discussed by Matthaei (2015), Matthaei et al. (2014), and Matthaei & Maurer (2015). The remainder of this chapter will show the status quo of the functional system architecture and present the enhancements compared to previous publications.

3.2 The Functional System Architecture

Vertical Abstraction Layers

Figure 3.1 illustrates a revised architecture based on Dickmanns et al. (1994), Dickmanns (2007), Matthaei (2015), Matthaei & Maurer (2015), Maurer (2000), Nothdurft et al. (2011), Reschka et al. (2011), Rieken et al. (2015), Saust et al. (2010), Ulbrich et al. (2014), Wille (2012), and Wille et al. (2010). The vertical abstraction layers of the functional system architecture are aligned to the levels of driving tasks from Donges (1999), Riemersma (1979), and Michon (1985, p. 498). One elementary (operational) stabilization layer is used for course keeping and speed control, a second (tactical) guidance layer is for any behavior planning and a third one is for strategic planning (navigation). Albus (1979, p. 281) suggested the use of such a hierarchical structure not only for behavior planning and control but also for perception. Nothdurft (2014) transferred the concept of Oberlander et al. (2008), to differentiate context information in particular for digital maps[5] by "topological," "semantic," and "metric" properties to the field of automated driving. In Figure 3.1, the terms road-level relate to the road network topology, lane-level to the semantic relations-

[4] Differences exist in the hierarchical separation of maps and localization and localization sensors. Similar as in the Stadtpilot project, there is no road-level environment modeling.

[5] Based on Oxford (2017a) and section 2.3, a map is understood as a diagrammatic representation of an area's scenery.

hips among lanes and feature-level to the metric properties used for a localization within a lane.

Certain modifications have been made by the team at the Institute of Control Engineering at TU Braunschweig after the publication of previous architecture versions in Matthaei & Maurer (2015) and Matthaei (2015, p. 37 ff.). The following sections describe the current state of the functional system architecture and discuss the recent modifications. It is explained along the inverted-"U"-shaped main signal flow through the components in the architecture.

3.2.1 Environment and Self Perception

Interfaces

The environment and self perception column has interfaces to localization and map provision, behavior planning and control, and with the communication column. Within the column there is an interface towards sensors.

Perception has an interface towards the automated vehicle's sensor systems. They have been clustered into environment sensors covering external aspects around the vehicle (exteroceptive) and vehicle sensors to obtain information about the vehicle itself and its internal state (proprioceptive). Environment sensors are sensors like cameras, lidar, and radar sensors but also conventional sensors like a thermometer or a rain sensor. Vehicle sensors provide internal information about the movement or pitch of the ego vehicle, but also information about the charging/filling level of the battery/fuel tank, for example. In a hardware architecture, sensor data feature extraction and even model-based filtering may be allocated to a sensor itself. Yet, in a functional system architecture, the interface between the sensor block and the subsequent feature extraction is raw sensor data. Sensors

Although, the perception column is primarily based on sensor data from within the column, it may use map information together with a pose within that map as input on different hierarchical levels of abstraction. On a macroscale level, there are topological road network maps used to augment perceived information with a-priori map information. On a mesoscale level, lane level map information may be used to augment context modeling even beyond the limited field of view from on-board sensor systems. On a microscale level within a lane, feature information may be used to provide additional landmarks or to stabilize lane tracking. Likewise an input might be Vehicle-To-X information obtained from other traffic participants or infrastructure. Inputs

The algorithms in the planning and control column are the primary data user of the perception column. On a navigation level, a road network together with a traffic flow may be used to calculate an optimal route. At the tactical level, a scene as defined in section 2.3 is provided. On an operational level, the perception may provide simple features and state variables as a low latency shortcut to low level control as in Maurer (2000, p. 42). Outputs

Perceived information is provided on different levels of abstraction (road-level, lane-level, feature-level) for map updates or mapping. Sensor data (gyroscopes, wheel tick sensors, ...) from the perception column may directly be used for localization and map provision.

Communication

Last of all, perception data may directly or indirectly be used for broadcasting information via Vehicle-To-X communication or visualization. The author assumes that there will always be a goal and value specific context selection and thus rather a for others as *relevant* classified situation subset to be broadcasted or visualized. Yet, also with this intermediate step, communication will be at least *based* on information from perception.

Comprised Activities

Colors and Distinctions

Figure 3.2 provides details on the environment and self perception. The dashed line symbolizes the separation between the perspective to the outside (environment perception) and the often neglected perspective to the inside (self perception) as in Maurer (2000, p. 58 ff.), Bergmiller (2015, p. 145 ff.), and Reschka et al. (2015). Similar as in Matthaei (2015, p. 51), a green color codes that only relatively certain internal information has been used. The blue color indicates that only internal sensors *and/or* environment sensor information has been used. The violet color indicates that additional map data with all possible errors in map-relative localization and incorrect, possibly outdated map information has been used, as well. The yellow color indicates perceived data used for map updates and Vehicle-To-X information.

Feature Extraction and Model-Based Filtering

Sensor data is used for feature extraction and subsequent model-based filtering. Feature extraction and model-based filtering is performed regarding several aspects. This includes lane detection and tracking, dynamic element tracking[6], occupancy grid modeling plus subsequent feature extraction and data filtering, traffic sign and traffic light recognition and state estimation, as well as self monitoring of the automated vehicle. Input to this block are raw or processed sensor data and possibly feature-level map data. Models are used to identify entities, associate measurements to entity hypotheses and track entities over time. In lane tracking, dynamic element tracking, and traffic light and traffic sign recognition a temporal validation or tracking is typically performed after an extraction of relevant features. In occupancy grid mapping, widely used for the stationary environment, a similar temporal filtering results from a probabilistic filtering performed in different cells of the occupancy grid itself. Entities and properties of these are generated by a subsequent feature extraction from that grid.[7]

Self Perception

Any of the sensors are mounted to the automated vehicle. Thus, their sensor data will be ego-relative. To transform sensor data into a stationary coordinate system, it is necessary to estimate ego motion. This is part of the data filtering in self-perception. The author suggests to comprise self perception further by a self monitoring. The threshold between a self monitoring and self representation on a context modeling level seems vague at first. The self monitoring provides informa-

[6] "Dynamic objects" form the set of "dynamic elements" by extending them with non-object-model-compliant elements (cf. section 2.3).

[7] The feature extraction and model-based filtering is not discussed in further detail here. Some details are discussed in Rieken et al. (2015) and will be discussed in a future publication specifically on this topic.

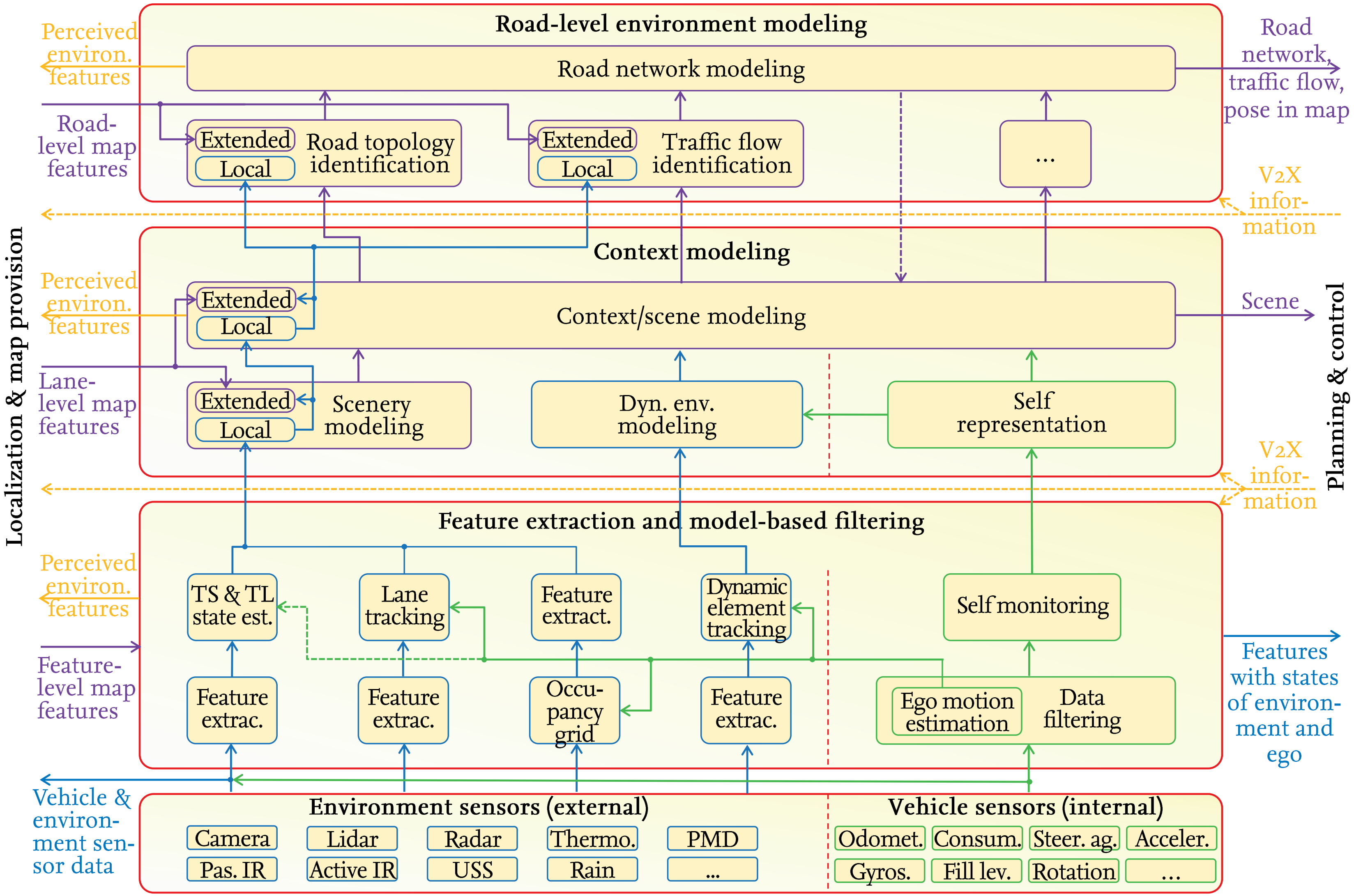


Figure 3.2: Environment and self perception based on Rieken et al. (2015) and Matthaei & Maurer (2015). Green = only subject to vehicle sensor errors; blue = subject to environment sensor errors; violet = also subject to map and localization related errors; yellow = perceived environment features and Vehicle-To-X information (dyn. env. = dynamic environment, TS & TL state est. = traffic sign & traffic light (recognition and) state estimation, extrac. = extraction, pas. = passive, IR = infrared sensors, USS = ultrasonic sensors, thermo = thermometer, PMD = photonic mixing device, rain = rain sensors, odomet. = odometry sensors, gyros. = gyroscope, consum. = consumption sensors, fill lev. = fill level sensors, steer. ag. = steering angle sensors, rotation = rotation sensors, acceler. = acceleration sensors, environ. = environment, V2X = Vehicle-To-X)

tion about entities of the ego vehicle and their attributes like health states or errors. The self representation provides semantic links between those entities to derive a full context not only about the environment but also about the automated vehicle itself.

Context/ Scene Modeling

The information from the feature extraction and model-based filtering is used for context/scene modeling (cf. section 2.3). This subsumes several aspects of information modeling, aggregation, and association. Scenery modeling combines lane information with a scenery model. This scenery model may use a-priori map data and a position in this map from the localization and map provision column in Figure 3.1. Dynamic environment modeling may interact with the scenery model to incorporate model-based information. Dynamic elements, for example, are more likely to move along lanes or paths.[8] Dynamic elements and the scenery are associated with each other to obtain an environment model. This is combined with the self representation of the ego vehicle to yield a context/scene model. This scene representation is transmitted to modules in the planning and control column. Matthaei (2015, p. 52) differentiates a "local" scenery and scene modeling from an "extended" one. The first is solely based on perceived information and incorporates no map-related information. Its output can be used for updating a map with perceived information. The distinction avoids loops in the information flow and self-confirming hypotheses of confirming map data with map-supported perception data.

Road Network

The perception column is completed by modeling a road-level environment. This subsumes a possible road topology and traffic flow identification to estimate which lanes constitute roads and whether these roads are congested or blocked.[9] So far, this module has not been implemented in the Stadtpilot or aFAS project. The road network is simply piped through as it is from an a-priori map from the localization and map provision column towards subsequent modules.

Enhancements to the State of the Art

Feature Extraction & Model-Based Filtering

The modifications are shown towards Matthaei & Maurer (2015) as the last broadly accessible publication of the functional system architecture in English. The sensors' block is identical; feature extraction and model-based filtering has only been marginally modified regarding the self perception. Here, Matthaei only mentioned the aspect of "motion estimation" and a rather vague "data filtering" (Matthaei & Maurer, 2015, p. 162; Matthaei, 2015, p. 51). Yet, as in Maurer (2000, p. 58 ff.), Bergmiller (2015, p. 145 ff.), and Reschka et al. (2015) this is only part of the self perception. It may further include, friction coefficient estimation, vehicle component wear-and-tear estimation, component diagnosis, energy level estimation, etc.

The aspect of traffic sign and traffic lights has marginally been modified. Matthaei & Maurer (2015, p. 162) called it traffic sign and traffic light "detection", Matthaei (2015, p. 51) called it traffic sign and traffic light "state estimation". Of course it is necessary

[8]For safety applications and to model non-rule compliant behavior, it is essential that this is only an information augmentation. The initial tracking results still need to be maintained to avoid crashing into non-rule compliant dynamic elements.

[9]A lane level traffic flow identification may still be considered as part of the context modeling.

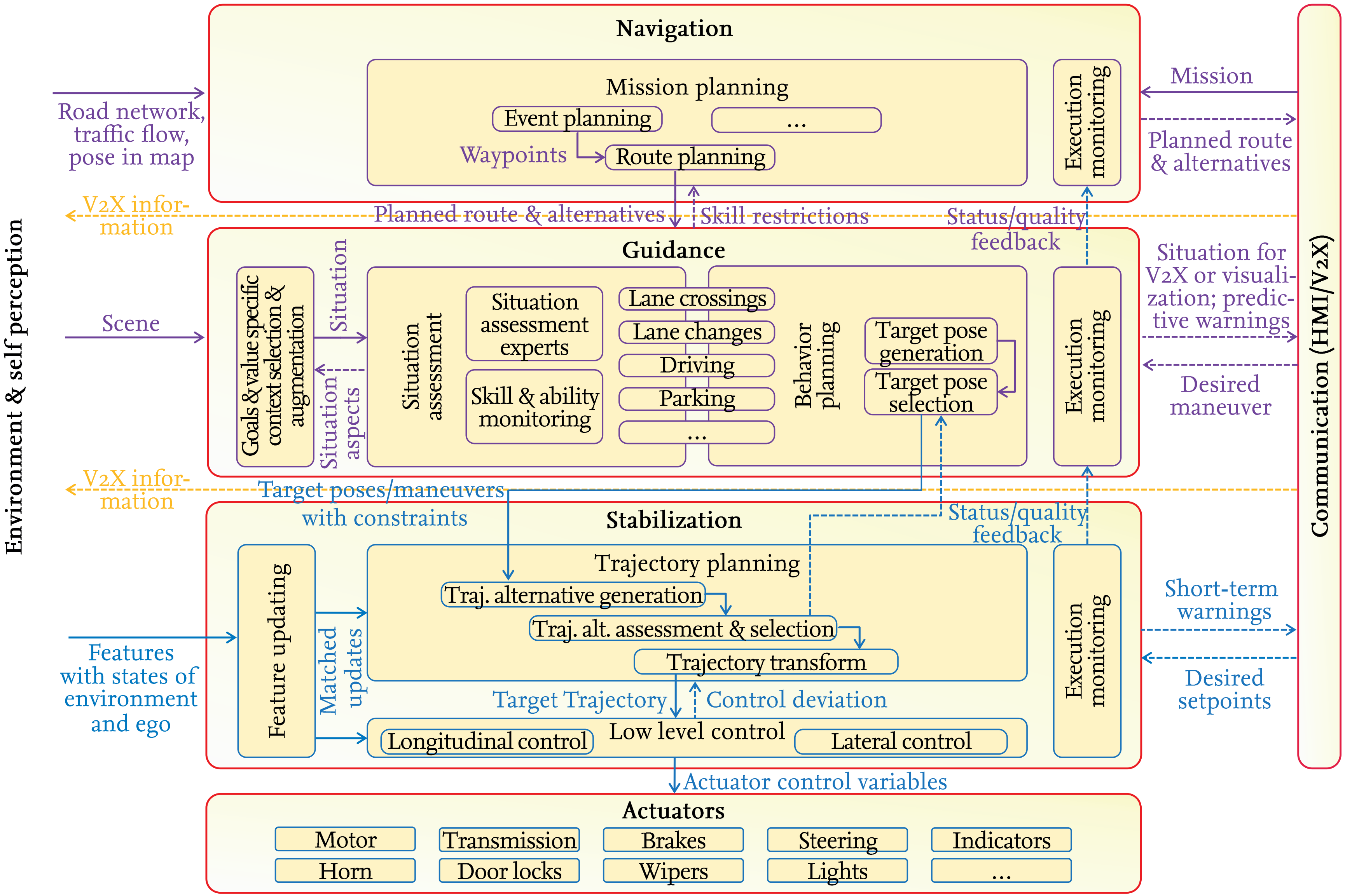


Figure 3.3: Submodules for planning and control. Blue = subject to environment sensor errors; violet = also subject to map and localization related errors; yellow = Vehicle-To-X information for perception (V2X = Vehicle-To-X, dyn. = dynamic, traj. = trajectory, alt. = alternative, HMI = human-machine-interface)

to detect, recognize the position, type and in case of traffic lights the state of an element. Other than tracking stationary lane markings/lanes, Matthaei assumes no need for an ego motion compensation for traffic signs and lights. Traffic sign and traffic light estimation has not been implemented in the Stadtpilot project. If this is purely frame-based, it may indeed not need an ego motion estimation. If it stabilizes traffic sign/light hypothesis over time, it will need an ego motion. Thus it has been linked by a dotted line.

Context Modeling

Context modeling has been restructured. Matthaei's differentiation between "local" scenery/scene modeling and "extended" scenery/scene modeling[10] have been both subsumed under only one scenery modeling and scene modeling with corresponding submodules. A dynamic environment modeling has been introduced as an analogon to scenery modeling for static environment aspects. This may include steps of validating different tracks of dynamic elements against each other. For instance if the contours of different elements overlap it might be a sign of actually tracking the same object twice rather than in fact observing a collision. Further, Matthaei & Maurer (2015, p. 162) and Matthaei (2015, p. 51) called the step of associating semantic information about the automated vehicle "vehicle state modeling". Aligned with Bergmiller (2015, p. 145 ff.), the author prefers *self representation* as a name for this block. Last of all, the name of the overall module seems odd at first. While it is named *context modeling*, its output is only a *scene* from a *scene modeling* as in Matthaei & Maurer (2015). With the definitions from section 2.3 it is indeed correct to have a scene as an output. Yet, the process itself entails aspects of context modeling, too. Thus the name of the module is extended to *context/scene modeling*.

Road-Level Environment Modeling

Similar to Matthaei (2015, p. 51), road topology identification and modeling as well as traffic flow identification are summarized in a block above context modeling. The block has been renamed from "road topology and traffic flow modeling" to a more general road-level environment modeling. Linguistically, this makes room to identifying and modeling aspects like ferryboats affecting the mission planning due to limited operating hours as a part of this block. Moreover, an arrow between road-level environment modeling and the context/scene model has been added to represent such an information flow of high-level road information towards information in a scene.

3.2.2 Planning and Control

Interfaces

Inputs

The planning and control column has interfaces to the perception and communication column and towards the actuators within the column. Inputs from perception are:

- A road network together with a traffic flow information for navigation.
- The scene described as in section 2.3 for tactical planning.

[10] This entails information from a-priori map data about static and movable elements.

- Features with state variables as a low latency shortcut to control as in Maurer (2000, p. 42).

Outputs

Outputs exist within the column towards actuators. These entail gas, gears, brake commands, and steering. Yet, it may also include actuation of other vehicle components like the horn, indicators or headlights. It may even include opening a door lock or the trunk for freight delivery or loading, or activating the wipers for removing dirt from the windscreen.

Interfaces towards the communication column will be detailed in its corresponding section.

Comprised Activities

Figure 3.3 illustrates details on the planning and control in a functional system architecture. The color coding for information flows is the same as in section 3.2.1. Modules for planning and control use the previously mentioned scene as a central interface on a tactical level. The modules have been divided into three levels according to the hierarchy of driving tasks in Donges (1999).

Navigation

On a strategic level, the road network, information about traffic flows or blockages and an externally provided mission are used for navigation purposes. A mission planning as in Dickmanns (2007, p. 405) or Gregor et al. (2002, p. 81 ff.) entails planning certain events like a cargo or passenger pickup. They result in waypoints between which a route needs to be planned. A route planning yields a – with respect to some optimization criteria – best route but also route alternatives.[11] The calculation of route alternatives may be triggered by events or upon request of the tactical level. The navigation may consider skill restrictions of an underlying tactical layer. If, for instance, if the battery of an electric vehicle is too low to take a shorter but more energy consuming route through a mountain area. Route alternatives to reach the mission goals are recalculated to reflect ego position changes.

Guidance

The guidance modules use this information to render a mission executable. They use the current scene to select relevant aspects and to augment it with additional information to derive one or several situation representations for the automated vehicle. Such a situation is used for situation assessment and behavior planning regarding several situation aspects. Among those are regular driving within a lane, lane changes, lane crossings (e.g., at intersections), free space navigation for parking, etc. (cf. “driving maneuvers [...] for automated vehicles”, Reschka, 2017, p. 122 ff.). Situation assessment for these situation aspects entails application specific situation assessment expert algorithms and also skill and ability monitoring for that particular situation aspects. Behavior planning entails not only maneuver selection but also planning about how a maneuver should be executed. This aspect of *how* does not include detailed velocity profile planning but rather a sequence of tactical behavior decisions. These could be longitudinal and/or lateral adjustments to a gap, stopping points in an intersection, indicator activations, or maybe even honking one’s horn. The guidance block is completed with execution monitoring of all com-

[11] The aspect of route alternatives has so far not been implemented in the Stadtpilot project or aFAS project. It has been implemented in the *Audi A7 piloted driving concept* vehicle.

ponents, which ensures reliability (continuity of correct service) and availability (readiness for correct service) (ISO, 2010). This execution monitoring has ultimate control over deactivating the system or its modules.

Interface between Guidance and Stabilization

Output of the tactical guidance layer is a set of target poses for maneuvers. A target pose commands the stabilization layer *what* to plan for. This may entail a target position, orientation, velocity (and further derivatives), constraints for trajectory planning like a drivable area, a reference corridor, sampling ranges or target deviation costs, and a symbolic maneuver type information.

The maneuver information may be utilized by an underlying stabilization level to switch between algorithms as in Maurer (2000, p. 74). The target pose may be linked to a vehicle with a certain id to perform longitudinal vehicle following. It may be set to the center of a neighboring lane for lane changing or it is set towards a gap in traffic for longitudinal adjustments to prepare lane changing. For parking, this pose may contain a goal position and orientation in a parking lot. Even at complex intersections, this interface is sufficient to implement, e.g., stopping at a stop sign, proceeding to a line of sight and finally turning through a lane with oncoming traffic.

Depending on the actual implementation only one or several[12] target poses may be handed over to the stabilization level. In case of the latter, target pose selection is implicitly done by the knowledge of selection rules in the stabilization level.

The stabilization subsumes trajectory planning and low level control, and execution monitoring as three major aspects. Trajectory planning calculates trajectory candidates for all these target poses. Low level control translates those trajectories into actuator control variables. Execution monitoring detects deviations between what is planned and executed.

Trajectory Planning

Trajectory planning as in Werling (2010), or path planning with a subsequent velocity profile planning as in Kammel et al. (2008), Hundelshausen et al. (2008), Wille (2012), Broggi et al. (2013) can be generalized into a three step procedure of trajectory alternative generation, trajectory alternative assessment and selection, and transforming the results into a representation to be used for low level control.

A path or trajectory planning may entail a subsampling of further target poses around the provided target poses as in Werling (2010, p. 42). Based on a cost function, the – to the cost criteria – best trajectory is selected.[13] Depending on the implementation of the trajectory planning, it is necessary to transform the trajectory from a geo-stationary, local coordinate system of a scene or situation towards an ego-vehicle bound coordinate system in which the actuators and low level controllers operate. If trajectory planning is executed in a Frenet frame, this transform is performed as a last step.

Low Level Control

A future point on this trajectory is used as input to the low level controllers to command a steering angle, brake pressure, or acceleration rate to the actuators

[12] So far, only one target pose has been implemented in the Stadtpilot or aFAS project.

[13] Selecting the best point could once more be considered as tactical decision making. Hence, one could argue the necessity of a trajectory selection arbiter block within the guidance module. For simplicity, it is excluded in Figure 3.3.

of the automated vehicle. To reduce latency, it may be necessary to obtain direct feature updates from the previously mentioned model-based filtering algorithms directly on the stabilization level. These feature updates may be incorporated into the low level controllers or even the trajectory planning.

Execution Monitoring and Feedback

Once more, the stabilization level entails execution monitoring to ensure the correct functioning of these algorithms and possibly to inform the tactical level about issues on the stabilization level. Examples of this driving task relevant information are if no collision-free trajectory can be calculated or if the execution of commanded behavior is not possible due to physical limitations in the vehicle's dynamics. This feedback is either used for execution monitoring in the tactical level or even to adopt the tactical behavior planning or strategic mission planning. For instance, if changing lanes to a highway exit lane jam-packed with traffic requires high relative velocity adjustments and thus high discomfort in trajectory planning, it may even affect the route planning by avoiding such maneuver and simply taking an alternate route by choosing a next exit further down the highway. Likewise, even low level control may provide such feedback by reporting control deviations. If a high slip angle indicates issues in vehicle stability, it may even affect tactical behavior planning by changing to a lane with better friction.

Enhancements to the State of the Art

Navigation

On the strategic level of navigation, the route planning has been renamed to a more general mission planning. When the scope of automated driving becomes wider, mission planning may not only contain route planning but even mission elements (Gregor, 2002, p. 43) like cargo pickup or refueling. Matthaei & Maurer (2015) mention a "selection of a next navigation point" as a submodule of the navigation block. Only transferring the next navigation point to a tactical planning imposes a severe limitation because several route alternatives may exist. This can be illustrated in the earlier mentioned example of an automated vehicle performing a lane change onto an off-ramp jam-packed with traffic. If there is a high risk to exceed the skills of the vehicle, it may be better to avoid such a risky lane change and accept a marginal detour rather than to enforce exiting where it was planned. This is not only a thought experiment but rather a real world issue and addressed by the lane advice in Ulbrich & Maurer (2015a). For that reason, the author deviates from Matthaei & Maurer (2015) by assuming not only one but several routes as an output of the route/mission planning and dropping the "selection of a next navigation point" altogether. Only if the alternatives are known, an informed tactical decision about following or deviating from what was planned at the navigation level is possible. Likewise to incorporate such knowledge about limited skills from a tactical level (either from the self representation as part of the scene) or the situation assessment and behavior planning itself into the mission planning, an additional upward facing arrow from guidance to navigation is added.

Guidance

Deep changes have been made to tactical planning compared to Matthaei & Maurer (2015). As illustrated in section 2.3 a goals- and value specific context selection and augmentation is added as an intermediate step between a goals- and value independent scene and a goals- and value related situation. There may be one or several situation data structures for different aspects of behavior planning. They can be used as an input or even be augmented by modules for situation assessment.[14] For instance, the results of a gap quality assessment might be fed back into a situation. That information could be used in an adaptive cruise control target pose selection module to temporarily reduce a time gap towards a front vehicle to avoid restricting gap adjustments to a gap slightly in front.

Behavior planning is used as an additional block to reflect not only a maneuver selection but likewise the earlier introduced planning about *how* a maneuver should be executed. The earlier introduced execution monitoring is added as an additional block to the planning and control column. No clear opinion has yet been formed if it is actually necessary to include execution monitoring as a separate block or if every block is supposed to have a sub-aspect of execution monitoring. Yet, as mentioned earlier, it is indeed important to include the *upward* information flow from stabilization to guidance.[15] It was missing in Matthaei & Maurer (2015) and has now been added.

Stabilization

The stabilization block has been detailed compared to Matthaei & Maurer (2015). A feature updating block has been added to reflect the updating process of, e.g., vehicle distances and velocities for low latency stabilization (cf. Maurer, 2000, p. 42). Trajectory target poses from the guidance level may be associated to dynamic elements. Their dynamic state variables may be updated based on more recent information directly from model-based filtering while bypassing the latency induced by the more comprehensive context modeling, situation assessment, and behavior planning. This leads to faster reactions in time critical scenarios.

Actuators

The set of actuators has been extended by adding indicators, the horn, door locks, wipers, lights, etc. Matthaei & Maurer (2015, p. 164) highlight that some actuators are used for the purpose of tactical communication (cf. "implicit communication" in Chapter 4). These actuators (or rather: devices) have not been part of the functional system architecture so far, neither as part of the communication column nor of the actuator block. Due to their similar nature as activating a brake light, they are all grouped under the actuator module. A module from the tactical level may actuate those devices *through* the operational level.

At last, the "planning and control" column has been renamed from the linguistically ambiguous term "mission accomplishment".

3.2.3 Communication

Interfaces

Strategic Level

The interfaces of the communication are illustrated in Figure 3.3. At the strategic level for navigation tasks, a mission may directly be commanded from an operator via a human-machine interface or even remotely via Vehicle-To-X communication. The mission may entail a route destination as well as goal criteria like a route with

[14] Other than Matthaei & Maurer (2015) the author prefers the less ambiguous term situation assessment instead of situation analysis. Yet, a situation is rather the input of a situation assessment than its output. Only some situation aspects may be needed for other modules in situation assessment and thus fed back into the situation data structure.

[15] This extension is based on discussions with Professor Chris Gerdes, Stanford University in 2014.

most comfort in automated driving, shortest travel distance or the most economic route alternative. As a feedback, the system may communicate a planned route, resulting from the commanded mission. Yet, the system may even provide route alternatives to an operator to enhance mission selection. The author agrees with Matthaei (2015, p. 56) that for a SAE level five system (cf. section 2.1) of an automated vehicle, the only necessary input is on a strategic level. Yet, for the sake of informing an operator or in case of not-level-5 systems, additional communication interfaces are necessary.

Tactical Level

At a tactical level for guidance tasks, a situation is used as an interface for visualization and Vehicle-To-X communication. While the situation for Vehicle-To-X communication may be different from the situation for behavior planning of the ego vehicle, it is still a situation because not every aspect that is part of the scene will be *relevant* for the (assumed) goals and values of any of the information recipients in Vehicle-To-X communication, or *legal* to be transmitted (cf. "enhancements" section). Likewise, a situation for visualization will probably be simplified and temporarily smoothed to reduce distraction. Yet, it is still a situation because it shows what is relevant regarding the goals and values of an operator or interested passenger. It may entail information about planned maneuvers as part of the situation aspects derived from planning and control. *Predictive warnings* to inform a passenger may either be considered as part of the situation or as a separate information interface from the guidance module towards the communication column.

In the opposite direction (towards perception and map provisioning), the communication column provides Vehicle-To-X information to be incorporated into the scene and possibly likewise on a feature or road level. Likewise, a desired maneuver may be commanded from an operator to the guidance module (Matthaei, 2015, p. 57). This could be to command an operator-initiated lane change but also to command an emergency stopping maneuver or a driver takeover request.

Operational Level

At the operational level, short term warnings may be issued or desired setpoints commanded (Matthaei, 2015, p. 57). Short term warnings could be the activation of an electronic stability control system in case of a higher than intended slipping angle on a low friction road. A desired setpoint could be the timegap towards a leading vehicle for an adaptive cruise control driver assistance system. For a future automated vehicle system these interfaces may not be necessary anymore, because by definition the system needs to handle all these aspects without driver intervention. Yet, as long as there is a transition between humans used to drive a vehicle by themselves and full automation these interfaces may still exist as a legacy for a long time.

Comprised Activities

An automated vehicle may have a communication interface for communicating with an operator or passenger (human-machine interface, HMI), as well as for technical communication with other traffic participants or the infrastructure via a Vehicle-To-X (V2X) communication interface.

The human-machine interface entails both directions of communication: On the one hand, to obtain input from an operator or passenger and on the other hand to

Human-Machine Interface

provide information. A special case are automated vehicles being monitored by a central tele-operation unit. Here the aspect of a human-machine interface and the usage of communication networks are combined. Matthaei (2015, p. 56) envisions the idea of strategic or tactical inputs for traffic management or clearing corridors for emergency vehicles. For the latter, the reliability and guaranteed coverage of current communication networks is an issue. Yet, at least the technically less demanding centrally controlled deactivation of an automated driving function within a certain amount of hours could be useful to ensure the absence of hazardous states caused by a bug, after such a bug has been discovered in the fleet of automated vehicles.

Vehicle-To-X Communication

The aspect of Vehicle-To-X communication entails communication with other traffic participants or infrastructure. Depending on what other vehicles are able to provide the range of applications is wide. Current research initiatives like Ko-HAF[16] address aspects like obtaining map updates from fleets, collaborative perception, and coordinating cooperative driving maneuvers among traffic participants. Algorithms to implement such behavior are spread among the modules in the other three columns of the functional system architecture. Yet, the actual communication interface for 802.11p wireless local area network communication, cellular network communication, or other communication channels is part of this column.

Enhancements to the State of the Art

Certain modifications have been made to the communication column since it was published in Matthaei & Maurer (2015).

Interface

Regarding interfaces, changes have been made to some contents of existing arrows. The interface between navigation and communication in Matthaei (2015, p. 57) is extended by not only exchanging a “route” but rather a “mission” as input to the navigation and by adding the aspect of route alternatives for the opposite information flow.

Vehicle-To-X Communication

While Matthaei[17] assumed collaboration happens over the interface left of the perception column, the author suggests to use the existing communication interface in the communication column. To the author, there is no need for a separate interface in the functional architecture, because aspects from the perception column can be exchanged with one interface at the very left. To allow an information flow from the communication column to the perception and map column, additional links have been added.

For transmitting Vehicle-To-X information, it is assumed that a full scene will probably never be sent but only a *relevant extract* of the aspects assumed to be relevant for the information recipients and their archival of their anticipated goals and values (situation for Vehicle-To-X communication). If little information exists about the goals and values of the information recipients, only obviously irrelevant aspects (e.g., privacy, what was seen inside of buildings by accidentally looking through windows) may be excluded and thus the *relevant extract* may almost converge against the

[16] http://www.ko-haf.de/, visited on Nov. 29th, 2016.
[17] Internal report “Cooperation, Collaboration, and Communication” from March, 2015.

full information from a scene.[18] If legislation and communication channel width will ever allow to broadcast a full scene, the aspect of information selection could be dropped and the link between the perception and communication blocks becomes bidirectional.

The localization and map provision column can exchange V2X information with the communication interface. Thus, the blocks in localization and map provision can receive and send updates of map data on all layers of the architecture.

3.2.4 Localization and Map Provision

Interfaces

The localization and map provision column has interfaces with the perception column to exchange:

- road-level map features and map updates,
- lane-level map features and map updates,
- feature-level map features and map updates, and
- vehicle and environment sensor data.

Further, it has an interface with localization sensors. According to Matthaei & Maurer (2015), the localization sensors like those in a global navigation satellite system (GNSS) are not part of the environment sensors at the bottom but are rather noted on the left due to providing information on higher abstraction levels. Localization Sensors

Within the column, information is exchanged between the different hierarchical levels. The upward information flow represents the use of, e.g., low level map features to extract higher level lane information. Likewise, there is an information flow downwards: Information about the existence of a road might be used to establish semantic relationships and support lane hypotheses in a lane level map. Within the Column

Comprised Activities

The automated vehicle needs to localize itself relative to its maps to make use of information in these maps. The aspect of map provision entails providing map information to other modules as well as the process of mapping and map updating in order to have such information to share. All these aspects are depicted in Figure 3.4.

Localization and map provision is executed on different hierarchical levels. Nothdurft (2014) transferred the concept of Oberlander et al. (2008) to distinguish map information by topological, semantic, and metric properties to the field of automated driving. Based on Du et al. (2004), Matthaei & Maurer (2015) differentiated between macroscale (road-level), mesoscale (lane-level), and microscale (within lane) map information and localization in those maps. Levels

[18] In the distinction between a scene and situation in Ulbrich et al. (2015g) the focus was rather on goals and values of a vehicle. Here the distinction has similarly been extended towards goals and values to be considered for communication as they are stipulated by authorities, e.g., privacy.

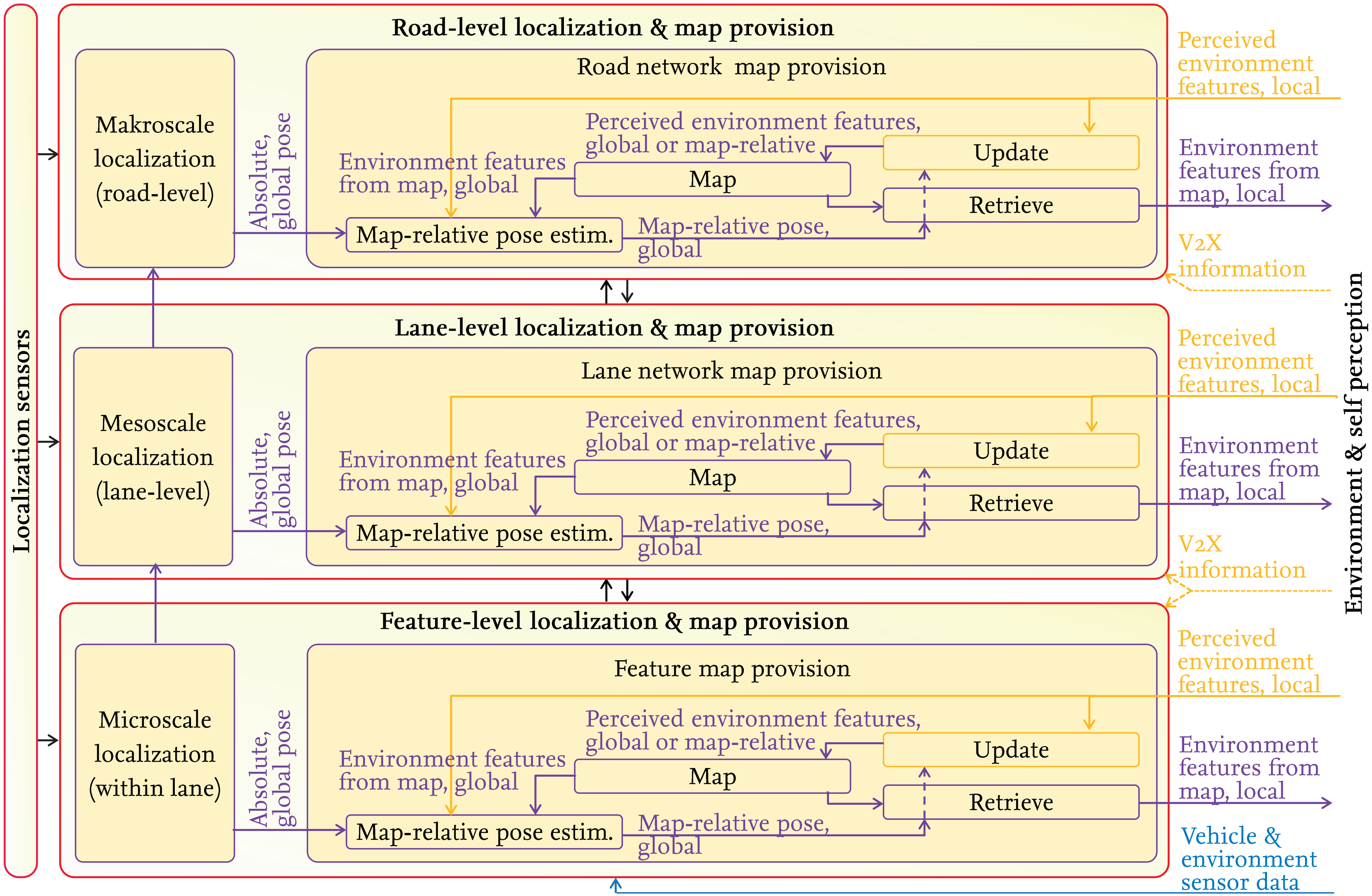


Figure 3.4: Localization and map provision based on Matthaei (2015, p. 45). Blue = subject to environment sensor errors; violet = also subject to map and localization related errors; yellow = features for map updates and Vehicle-To-X information (estim. = estimation)

On each level, localization sensors provide data input to obtain an absolute, global pose from localization algorithms. This information is often combined in Bayesian filtering approaches with inertial movement data (cf. blue data flow in Figure 3.4) to provide a position even between position fixes from, e.g., a satellite-based localization sensor. Current approaches are differentiated by their depth of data fusion (loosely, tightly, ultra-tightly-coupled) and summarized by Skog & Händel (2012) cited by Matthaei (2015, p. 43).

Localization

An absolute global pose is used together with perceived environment features to obtain a map-relative pose estimation. This map-relative pose is used to *retrieve* map information and to provide it to modules in the perception column in order to augment perceived information by map information.

Retrieval

Depending on the implementation, a second data flow from the perception column towards the map provision column may exist. This is to use features and a concurrently obtained map-relative pose to update maps with perceived information. This concurrent map-relative localization and map-updating process may be repeated on the earlier introduced hierarchical levels. Information may be exchanged between the levels to keep maps consistent.

Updating

Different technologies exist to serve the different (vertical) levels in the functional system architecture with different needs for accuracy. On a macroscale level (roads), global navigation satellite system solutions found in today's vehicle entertainment systems are largely sufficient. For mesoscale localization on a correct lane as well as for microscale localization within a lane, a higher accuracy is needed. Signal distortions in ionospheric layers can be compensated by utilizing different carrier frequencies and correction data from ground stations may be used to increase accuracy. Yet, accuracy as well as reliability are yet insufficient to serve as a single, non-redundant source for localization in automated driving. This becomes particularly obvious in urban environments or complex multi-level highway interchanges.

Limitations

All information from the localization and map-provision column is subject to errors in the localization as well as errors in the maps itself. At the time of writing there is no guarantee on information integrity and timeliness of data. Thus, incorrect localization or map data may possibly propagate to subsequent modules and compromise decisions and behavior. To ensure awareness of this, every module that uses map data is colored in violet.

Error Propagation

Enhancements to the State of the Art

The localization and map provision has been restructured. The "external data" and "absolute global localization" columns in Matthaei & Maurer (2015) have been summarized into one "localization and map provision" column. "External data" was renamed to map provision to ensure that modules are activities, as in the Unified Modeling Language (UML) standard. Hence, "external data" is – similar to, e.g., a "scene" – a data container and thus an arrow rather than a module. The author thinks that the level of abstraction for "localization" and "external data" seemed less aggregated than for example "perception" or "planning and control" which form other columns. Further, the titles of the map provision blocks has been changed towards what they actually provide: Maps. The "world modeling" used by Matthaei & Maurer (2015) leaves room for confusing it with the term's connotation in the community to where it reflects activities summarized under context modeling. Matthaei & Maurer (2015) did not provide details within the map provision block. The refinements in Matthaei (2015, p. 45) within those blocks have now been incorporated to make them accessible to a non-German-speaking audience.

3.3 Open Issues

Despite long and intense discussions, there are still several open issues in the functional system architecture. These aspects will be highlighted here:

Context Modeling

First of all, the name of the context modeling seems counter-intuitive due to the fact that it only outputs a scene. Indeed, a scene is part of the context and according to its wide definition (cf. section 2.3), a full context may never be represented. Yet, certain context information may be used for better scene modeling. Thus, the term "context modeling" for the overall block seems more appropriate.

Additional Links

Secondly, in architecture discussions with other research groups in the Uni-DAS society[19] the idea was voiced for feedback from stabilization modules towards model-based filtering modules. That is, to adopt models if, e.g., the ego vehicle is not following a planned trajectory when it is drifting.

Driver Monitoring

Thirdly, no clear answer has yet been provided where a driver or passenger[20] monitoring camera should be located in the architecture. One could argue that it is irrelevant if an operator provides a maneuver input by a button or the camera and that it thus shall be part of the human machine interface. Likewise, it may be considered as a sensor and part of the perception column. A driver or passenger monitoring is so far not part of the Stadtpilot project or "Jack".

Grid Maps vs. Feature Maps

Further more, an open point is the clear differentiation between "occupancy grid mapping" in the perception column and "feature map provision" in the localization and map provision column. Occupancy grid mapping is necessary in perception for local dynamic maps, free space extraction, or dynamic classification. If static elements are aggregated in a global feature map, it is part of the map provision column. Hence, the age of features to be typically still maintained in the grid or map is a distinguishing factor, but there is still room for a better distinction between both.

Shared Maps

In current discussions about the potentials and demands of automated vehicles, a server-based shared map is a key to the availability for automated vehicles. It is not explicitly modeled in the architecture, since the author assumes it to be part of the V2X connectivity. A more sophisticated integration into the architecture of the ego vehicle seems not helpful, as it would change the focus from the aspired architecture for a single automated vehicle towards an overall architecture for a

[19] Uni-DAS workshop on functional system architectures in October 2015 in Darmstadt, Germany. www.uni-das.de

[20] In a SAE level 3 to 5 system, it might be necessary to help minors or elderly passengers for instance in case of a medical emergency situation or to ensure that they remain seated while driving.

whole traffic system. That would require several additional aspects like trusted authorities for information validation or traffic management authorities, which are out of scope of this chapter.

Moreover, there are still discussions on the point whether navigation or guidance has ultimate decision power if a planned route is followed or an route alternative is selected. If a traffic jam is detected, it is clearly a navigation task to adapt the route. Vice versa, if enforcing to take a highway exit would result in a collision, it is the tactical layer that decides to not take the exit and to request a replanned route to reflect the reality of having missed that particular exit. There is a gray area in between where following the route is still within the specifications of what the automated vehicle *can do*, but where in the given situation it is *just now*, tactically a better choice to rather pick a route alternative with a minimal detour to avoid risk or maintain comfort goals. As in section 3.2.2, the author sees these decisions to be under the decision-making authority of the guidance level, but not without controversy.

Ultimate Decision Power

Another issue is where predictions are to be found in the architecture. To the author, a prediction is rather a tool to be used in several modules. For instance, model-based filtering will use prediction models. Likewise, a situation prediction might be necessary in the guidance module or a movement prediction in the stabilization module. One could ask if there is a prediction even in the context model to provide not only the current but also future scenes. A possible way to illustrate predictions in the architecture could be to extend the two-dimensional architecture by a third dimension in which prediction is an additional layer. This comes to the price of visual distinctiveness and presentability. Another way could be to introduce multiple *views* on the architecture for particular aspects.

Predictions

Further more, the allocation of self representation to a particular block in the architecture is not as clear as it seems. For sure, it is mainly a bottom up process to aggregate information from vehicle sensors. Yet, execution monitoring might detect that a vehicle's deviation from its intended trajectory is high and thus the maneuver capabilities of that vehicle are limited. In other terms, there is goal and value specific information for self modeling in the planning and control column. Hence, certain aspects of self modeling could be spread over several hierarchical levels and columns in the architecture and thus limit the conceptual rigorousness that structure diagrams of the architecture suggest. Once more, a third dimension with a separate layer for self representation could alleviate this issue. In this layer not only the self representation, but also all forms of self monitoring and execution monitoring could be placed. The result could be aggregated in the scene/context model and used for decision making and control in the planning and control column.

Self Representation

Possibly not fully covered is the aspect of cooperation and competition between multiple agents. So far, implicitly cooperative behavior (Ulbrich & Maurer, 2013) and explicit Vehicle-To-Infrastructure communication (Saust et al., 2012) has been implemented in the Stadtpilot project. Yet, it seems likely that future research on cooperation and competition may not be fully covered in the architecture. The

Cooperation and Competition

author assumes an additional *view* on the architecture might be required to cover these aspects with all its various facets.

Vehicle-To-X Communication

Last of all, the role of Vehicle-To-X communication is still subject to discussions. While the current communication column is eligible to broadcast information from the planning and control or perception column, an opposite communication flow for Vehicle-To-X data input is harder to incorporate. Currently, this induces a right-to-left information flow that contradicts the main signal flow direction otherwise going from left-to-right. A workaround would be once more to open a third dimension or additional *view* for Vehicle-To-X communication as it has interfaces with many blocks. A possible implementation specific addition to the architecture could be a data flow from the decision modules to other traffic participants or the infrastructure via V2X and vice versa. E.g., the selected route, the selected maneuver as part of the situation for Vehicle-To-X communication, or a planned trajectory on the stabilization level.

3.4 Conclusions

This chapter presented a refined functional system architecture for an automated vehicle. The concept of hierarchy and functional separation has been introduced and applied. The interfaces between the modules have been detailed and the modifications to the state of the art have been presented. Behavior planning as the core of this thesis has been presented in the overall context of the functional system architecture. To the author, this functional system architecture is still an organic structure that will be modified and refined to address the open issues.

4 Cooperative Driving and Cooperative Behavior Planning for Lane Changes [1]

The issue of cooperative driving and behavior planning has come into focus in the automated driving research community. While it is already difficult to come up with a definition of what cooperative behavior and cooperative driving actually is, it is even harder to contrast it with what has been done in previous decades in the field of tactical behavior planning.

Links & Structure

For a systems perspective, cooperative behavior is central for tactical lane change behavior planning. It extends the focus from an automated vehicle itself towards a group of interacting vehicles or even the impact on the traffic system resulting from these interacting vehicles. Without considering cooperative behavior, it may never be possible handle inter-vehicle interactions sufficiently well to fully substitute a human driver. This chapter utilizes the terminology from Chapter 2 to differentiate cooperative behavior by cooperative skills and abilities and communication and awareness channels. The discussion is substantiated by identifying scenarios for cooperation, which will be addressed by the reward model in section 10.5.1 for lane change planning.

4.1 Structuring the Issue of Cooperation

Cooperation in Psychology

In psychology, Spieß (2014) defines cooperation as a form of societal collaboration between persons, groups, and institutions, or respectively as social interaction. Spieß stresses that cooperation entails conscious and planned acting as well as processes of mutual coordination regarding specific objectives. Cooperation is based on fair conditions of collaboration and reciprocity.

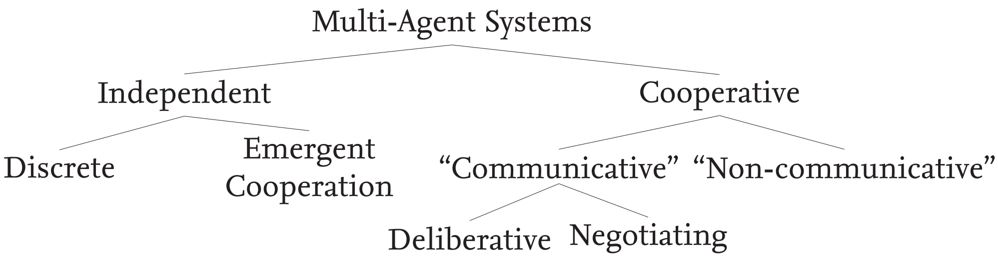


Figure 4.1: Cooperation typology based on Franklin in (Doran et al., 1997)

[1] Parts of this chapter have been pre-published by the author in Ulbrich et al. (2015f). The coauthors contributed a review and discussions. In particular, they added the aspect of the “two levels of cooperation”, helped to implement key aspects of scenario SC 6 and SC 7 from Table 4.1, took care of bugfixing, and contributed test-cases for several of the scenarios in Table 4.1.

Cooperative Multi-Agent Systems

Franklin (Doran et al., 1997) identifies *cooperative systems* as those in which the agendas of the agents explicitly include cooperation with other agents. He further differentiates between *communicative* and *non-communicative* systems. Communicative multi-agent systems *intentionally* send and receive signals to and from other agents. Among those, he subsumes *deliberative* and *negotiating* systems. The first jointly plan their actions, while the latter extend the former with the idea of competition for resources. Non-communicative agents coordinate their cooperative activity by observing and reacting to the behavior of others. As Watzlawick et al. (1967, p. 48 ff.) pointed out that "one cannot not communicate"; the author suggests using the terms *explicitly communicative* and *only implicitly communicative* instead of "communicative" and "non-communicative".

According to Doran et al. (1997), cooperation occurs if the actions for each agent in a multi-agent system fulfill at least one of the following two conditions:

> "(i) The agents have a (possibly implicit) goal in common (which no agent could achieve in isolation), and their actions tend to achieve that goal.
>
> (ii) The agents perform actions which enable or achieve not only their own goals, but also the goals of agents other than themselves."

Doran et al. (1997) further differentiate cooperative group processes by the degree of altruism on a scale from purely self-interested cooperation to wholly altruistic.

Norman uses a much clearer definition of cooperation in (Doran et al., 1997). He defines that cooperating means "to act with another or others for a common purpose and for common benefit". This definition embraces three aspects: The aspects of *acting together*, a *common purpose*, and a *common benefit*.

Implicit vs. Explicit Cooperation

The DFG priority program proposal by Stiller et al. (2013, p. 5) differentiates between using *implicit* cooperation *only* and using *implicit together with explicit* cooperation. The program proposal avoids a clear definition of explicit and implicit cooperation. Rather it illustrates the differentiation using two examples: According to Stiller et al. (2013, p. 1) explicit maneuver coordination makes it possible to plan driving trajectories within safety critical margins, for which human drivers are not able to do the same due to their limited communication and reaction abilities. Hence, explicit cooperation seems to necessitate explicit communication. As an example for implicit cooperation a merge situation is discussed where the driver's head poses or even the lateral offsets within a lane are used to negotiate the merging process. In Matthaei et al. (2015), the author's team pointed out that Vehicle-To-X (V2X) communication is just a particular communication channel among other channels that have been used long before, such as indicators, flashing headlights, or a signal horn. The author is not aware of any conclusive argument on whether those signaling devices should be considered as explicit or implicit communication.

Two Cooperation Levels

Lacking a clear definition of implicit and explicit and possible shades in between, the authors in Matthaei et al. (2015) defined two levels of cooperation. The first subsumes "compliance with current traffic laws" like basic collision avoidance or rules to be followed for traffic flow control, e.g., at intersections or in highway merging maneuvers. The second level subsumes any means for more sophisticated utility-optimization, e.g., leaving a gap towards a preceding vehicle in order to allow another vehicle to merge in front of the automated vehicle more easily. Thus, the main difference between both levels is the degree of altruism and consideration for the overall system utility instead of the individual utility. Traffic laws define a certain minimal level of cooperation (first level). To achieve mission goals quickly and efficiently it might be advisable to cooperate even on that second level.

Cooperative as an Attribute

Düring & Pascheka (2014) clarified that agents are not "cooperative" or "not-cooperative" per se, but rather that *cooperative* is an attribute of an agent's behavior. They define "cooperative behavior with respect to [another] agent [...] and with respect to a total utility function [...], if by choosing this behavior [the first] agent [...] knowingly and willingly increases the total utility [...] in a coupled situation, compared to a reference utility." They assume that knowingly and willingly shall imply more cooperative behavior than just what the agent is *forced* to do "by legislation or physical laws". Thus, it seems that they prefer a narrower definition of cooperation that is only limited to the second level of Matthaei et al. (2015).

Figure 4.2 illustrates the different levels of skills and abilities and existing communication channels to address these. It lists examples for these situations in daily driving situations. It differentiates between disparate channels of communication and awareness and different levels of cooperative skills and abilities.

Channels of Communication and Awareness

As indicated previously, different communication and awareness channels exist. Apart from technical channels such as a Vehicle-To-X (V2X) communication interface, a standard vehicle already has – by traffic laws mandatory – signaling devices for communication (indicator, horn, etc.). Moreover, intentional gestures like longitudinal driving maneuvers, pedestrians' hand gestures, or even a driver's hand gestures might be used. Technically challenging but still a viable mode of communication include unintentional gestures. All communication channels can be used to receive or transmit information. E.g., human drivers can intentionally use indicator flashing to communicate an intention and/or be able to perceive the indicator flashing of other vehicles. If the receiving part dominates the usage of the channel it may be more intuitive to call it an awareness channel. All channels necessitate the detection and tracking of objects to cooperate with them.

Cooperative Skills and Abilities

Any kind of cooperative multi-agent systems in the scheme of Franklin in (Doran et al., 1997) require certain cooperative skills and abilities, e.g., regarding perception, reasoning, or communication. These skills and abilities enable cooperation on different hierarchical levels and allow different scopes of optimization and altruism as highlighted above.

Examples of Cooperative Behavior

Cooperation skills and abilities are to some extent hierarchically distinguished. On a very basic abstraction level, cooperation can be achieved by the communication and consideration of states and actions for cooperative driving behavior. A possible state to be communicated could be an intervention of the Electronic Stability Control (ESC) system of a vehicle. A possible action to be communicated could be a braking action. The next level does not address single actions or states, but rather tactical maneuvers. A maneuver entails a sequence of actions and states. Maneuvers, which are to be executed are motivated by intentions and goals. Thus, an even higher abstraction level is to communicate and consider the intentions of

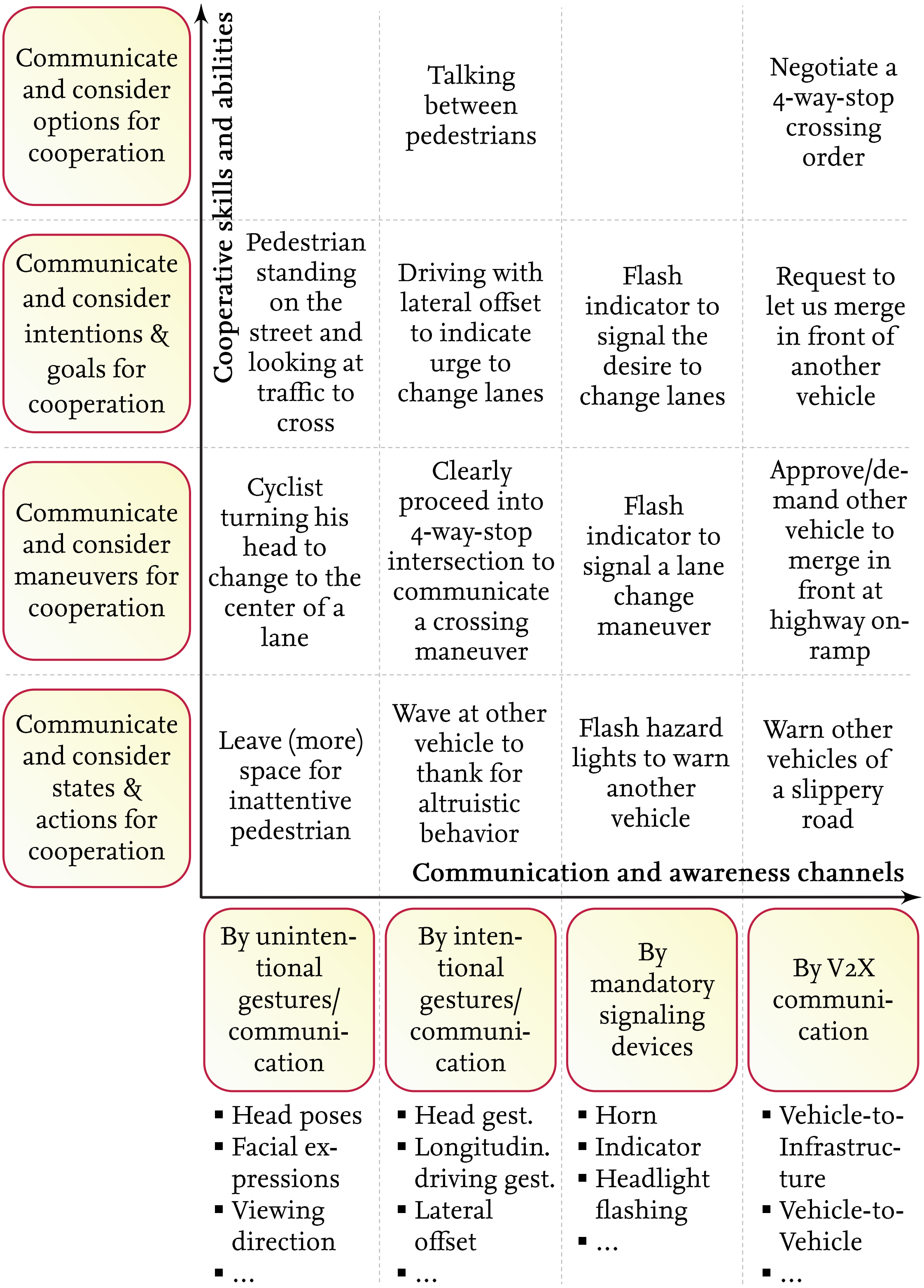


Figure 4.2: Examples for cooperation classified by communication channel and necessary cooperation skills and abilities (gest. = gestures, longitudin. = longitudinal)

other vehicles even before they are substantiated into executable maneuvers. An example of such an intention could be a bus at the roadside communicating its merge intention and an altruistic automated vehicle which brakes a bit for the bus and communicates that it will let the bus merge back into the traffic. If the skills and abilities are limited by the communication and consideration of intentions and goals, decisions need to be made locally by each *deliberative agent* (cf. Figure 4.1). Vice versa, for *negotiating multi-agent systems* it is necessary to communicate (inquire and answer) and consider the *options* for behavior alternatives. It is essential to negotiate by exchanging information about the costs of potential behavior options (intentions, goals, maneuvers, actions). This makes it possible to achieve solutions that are closer to the true overall system optimum with yet independent agents. As an example of this, it could be beneficial for a bus to communicate via V2X high costs for the behavior option of not letting it merge in front of an automated vehicle because the bus is behind schedule.

Communication about Options

It is to be noted that this communication about options rarely happens in today's traffic. On the one hand, this is because typical traffic participants do not have a communication channel with a suitable bandwidth (shouting between fast moving vehicles that one is behind schedule is rarely seen). On the other hand, this level of cooperation requires honest behavior. An individual agent gains personal benefit from communicating wrong behavior option costs. Without a mechanism (e.g., a system of trust) to penalize such selfishness, cooperation on such a high abstraction level might not work at all. Among the few examples for cooperation on such a high level are situations where people communicate directly between each other, e.g., about missing a flight, if they have to wait in a long queue at an airport security check facility. Knowing the costs of the options makes the discomfort of longer waiting bearable against causing another individual to miss his or her flight. The cooperation only works because of a certain level of trust that the seemingly belated person is not lying.

4.2 Open Issues

This section highlights issues regarding cooperative behavior for automated driving.

Issues of V2X Communication

Figure 4.2 provides a scheme to substantiate the vague term of "cooperation". Yet, Figure 4.2 also illustrates one of today's main challenges for cooperative, automated driving. On the very right it lists possible cooperation scenarios using Vehicle-To-X (V2X) communication. However, since V2X communication is not widely available as of 2017, it is at best an additional channel to improve comfort if other vehicles are able to use it. Moreover, cooperative automated driving also needs to work without this channel. As the availability of V2X communication partners is close to zero for a regular stretch of highway or urban area driving, no particular attention is given to cooperation scenarios with V2X communication in this thesis.

At the other end of the axis of communication and awareness channels is the communication by intended and unintended gestures. Using these channels imposes high perception requirements, which are currently not fully met. As of 2017 it is

Issues of Communication by Gestures

already challenging to perceive lanes and decently sized objects like cars or trucks reliably. Identifying the driver in a tracked car is at the forefront of research. Estimating a head pose or hand signals seems out of reach for reliable use for today's cooperative, automated vehicles. The same is true – although to a lesser extent – for detecting indicators, brake lights, or a headlight flashing.

Cooperation vs. Regular Behavior Planning

In essence, cooperative automated driving is limited to very few situations, for which communication and awareness channels exist in today's automated vehicles. For many of those, it is rather a matter of preference to call them *cooperative behavior* or just regular automated driving. E.g., aborting an already initiated lane change maneuver due to suddenly perceiving a vehicle on the neighbor lane could be considered a cooperative maneuver to prevent discomfort or stress for the other vehicle, but it could also just be called a necessary basic feature of a lane change planning module.

4.3 Scenarios for Cooperation

Addressable Scenarios

As a consequence, it is worthwhile to ask what scenarios can be solved with today's "cooperative" behavior skills and abilities with current environment perception power and without omnipresent V2X equipment. Table 4.1 shows a list of addressable scenarios for lane change behavior planning and conjoined cooperation.

Regular Lane Changes

The first scenario in Table 4.1 is to consider the (dis-)comfort costs of other vehicles during lane changes. Cooperative behavior is more obvious in the second scenario SC 2. It illustrates giving way for a pressing rear vehicle by temporarily accepting a certain disadvantage for the ego vehicle, to give another vehicle the advantage of driving at its own pace. This is particularly relevant in domains where overtaking on the right is prohibited. The inverted scenario of requesting cooperative behavior by tailgating a slow front vehicle is illustrated in the next figure. Depending on the road traffic rules it may not be legal.

Longitudinal and Lateral Maneuvers

Another form of cooperation is the communication of driving intent by longitudinal and lateral maneuvers as in scenarios SC 4 and SC 5. It illustrates the usage of a lateral offset to communicate a lane changing intention. Likewise, longitudinal adjustments to gaps may be used to communicate into which gap a lane change is intended.

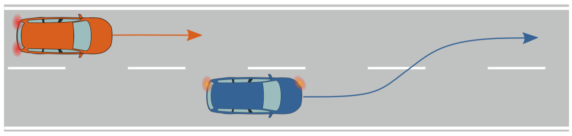
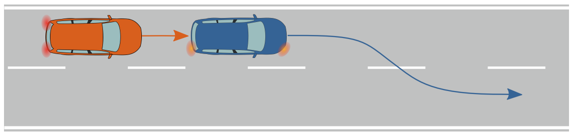
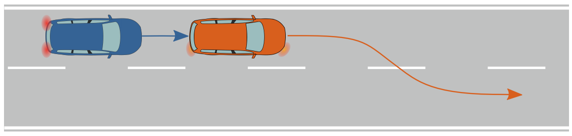
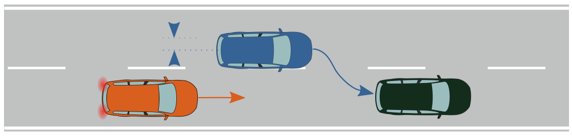
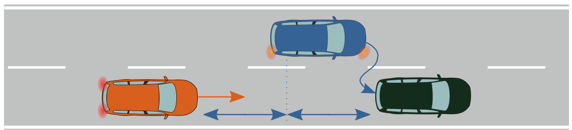
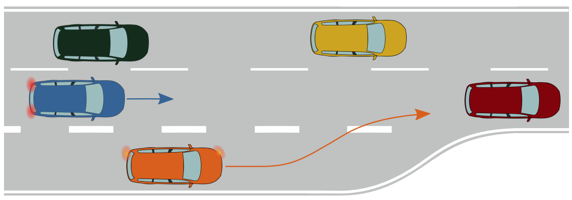
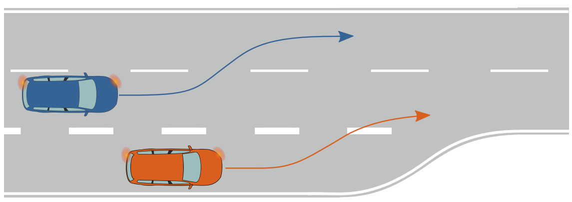
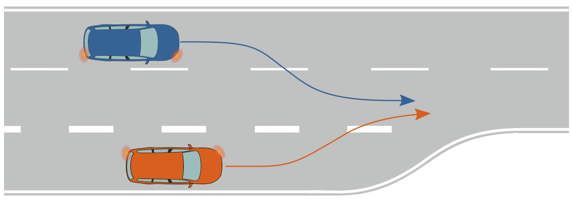
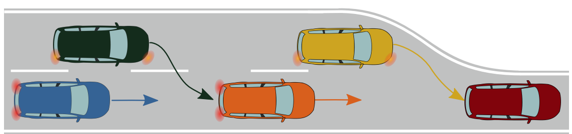
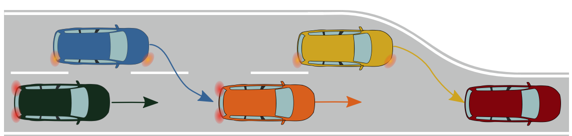
Merging

Scenarios SC 6 to SC 10 illustrate cooperative behavior in merging scenarios. Scenarios SC 6 and SC 7 illustrate cooperatively giving way to a merging vehicle by executing a lane change or braking. Scenario SC 8 extends the previously mentioned scenarios by avoiding a lane change into a lane which will be occupied by a merging vehicle. Scenarios SC 9 and SC 10 illustrate cooperative merging and letting merge in zipper method merging areas.

4.4 Conclusions

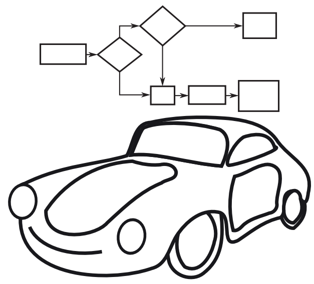
This chapter identified two dimensions to differentiate cooperation. On the one hand, communication and awareness channels are used to distinguish cooperative behavior and on the other hand the hierarchical level of cooperative skills and

Table 4.1: Currently addressable cooperation situations for lane changes (ego vehicle in blue)

Nr.	Scenario	Illustration
SC 1	Considering (dis-)comfort costs for rear vehicles	
SC 2	Giving way to pressing rear vehicles	
SC 3	Requesting cooperation of slow front vehicle ("tailgating"). May not be legal	
SC 4	Squeezing into gaps by lateral offsets to the lane center	
SC 5	Squeezing into gaps by longitudinal adjustment to gaps and usage of the indicator	
SC 6	Letting vehicles merge in front if their lane or on-ramp ends soon, or if they enter the traffic from a bus stop or parking space	
SC 7	Clearing a lane for vehicles if their lane or on-ramp ends soon, or if they enter the traffic from a bus stop or parking space	
SC 8	Not changing to a lane on which another vehicle is about to merge to	
SC 9	Dedicated handling of zipper method merging where the automated vehicle is letting merge	
SC 10	Dedicated handling of zipper method merging where the automated vehicle merges	

abilities. A matrix is formed and typical cooperation tasks are segmented into that matrix of both dimensions. Last of all, scenarios for cooperation in tactical behavior planning for lane changes are identified and described. They serve as a basis for the implementation of cooperative behavior in section 10.5.1. Finally, this chapter challenges the role of cooperative behavior in today's automated vehicles. Given the current low penetration rate of Vehicle-To-X communication partners and the very limited perception skills for detecting intentional and unintentional gestures, possible cooperative behavior is still limited.

5 Frameworks for Tactical Behavior Planning



The essence of this thesis will be to improve behavior planning for tactical maneuvers. This chapter reviews different approaches for doing this. First of all, this chapter will relate the issue of decision making for behavior planning for a technical system to a more general framework of human decision making. To implement decision making for a technical system, special attention will be paid to probabilistic behavior planning methods because driving decisions in automated driving are inherently uncertain.

5.1 Phases of Decision Making for Behavior Planning

The Rubicon Model

In cognitive psychology, the Rubicon model of decision phases has been established by Heckhausen & Gollwitzer (1987). According to Figure 5.1, behavior planning begins with a predecisional deliberation phase, in which a “commitment to a specific [...] goal intention” is formed based on a situation-person (agent) interaction (Achtziger & Gollwitzer, 2008, p. 274). Once this “Rubicon” has been crossed, a human/agent “chooses strategies and formulates plans [...] conducive to attaining the aspired goal” in the “volitional preactional (or: postdecisional)” phase (Achtziger & Gollwitzer, 2008, p. 275). The transition from a motivation to a volition[1] “indicates that the motivational deliberation of potential action goals has been terminated [...]” (Achtziger & Gollwitzer, 2008, p. 275). With the transition into the actional phase, developed plans are executed. An intention is deactivated after an action has been completed. In a postactional evaluation the actual result of the action is compared to the intended outcomes/goals and consequences may be derived.

Transfer to a Technical System

The basic structure of the decision problem remains the same, regardless of whether a human or a technical system is making the decisions. Similar to a human, an automated vehicle needs to form an intention, develop a planned behavior to render an intention executable, and finally execute this behavior.

Challenges

However, the last step of evaluating actions, deriving consequences, and possibly learning from it requires high cognitive skills. Moreover, this last step may change the decision behavior over time and is thus difficult to validate for functional safety considerations. Therefore, this last phase in the Rubicon model will be ignored for the remainder of this thesis.

[1]Based on Lewin (1926) and Ach (1935), volition is understood as “the form of motivation involved in goal striving”, thus the “translation of existing goals into action[s] and [...] the regulation of these processes” (Achtziger & Gollwitzer, 2008, p. 276). Vice versa, motivation is used to address “the motivational processes involved in goal setting” (Achtziger & Gollwitzer, 2008, p. 276).

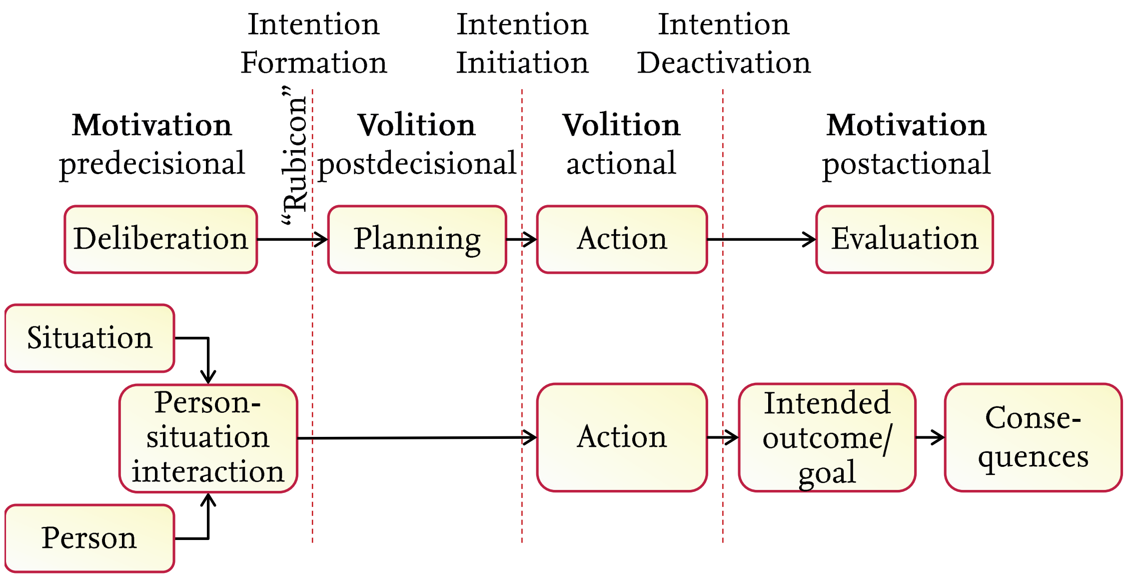


Figure 5.1: Rubicon decision phases (Heckhausen & Gollwitzer, 1987) and cognitive motivation model (Rheinberg, 1989, p. 104) combined as in (Heckhausen & Heckhausen, 2008, p. 7)

Rubicon in Technical Systems

Another issue is where the "Rubicon" is actually for a technical system: The author agrees with Achtziger & Gollwitzer (2008, p. 273 ff.) that motivation and volition should be differentiated into distinct processes of deliberation and planning. Yet, to the author, it is less relevant for a technical system, if an intention is formatted but not initiated, because a technical system will probably never feel *unsatisfied*. Furthermore, a technical system may be able to reason about several behavior plans *at once*. Therefore, crossing the "Rubicon" by transforming a motivation into a goal may be less of an issue for a technical system. Vice versa, the intention initiation may even be a bigger "Rubicon", because at this transition others will actually *see* if a behavior is inconsistent.

5.2 Probabilistic Behavior Planning

Uncertainty and Risk in Decision Making

Decision making under uncertainty (cf. section 2.5) does not only apply to technical systems. In fact, any human driving a car faces exactly the same challenges. In economics (Knight, 1964, p. 19 f.) and in cognitive psychology (Gigerenzer, 2014) it is common to differentiate between decision making under *risk*[2] and decision making under *uncertainty*. Decision making under risk means all options, consequences and probabilities are known or can be estimated. Typical approaches are known from game theory. If not all of these conditions hold true, this is called decision making under uncertainty.

Heuristics

Decision making under uncertainty requires decision making heuristics (Gigerenzer, 2014). A central problem for real world robotics is to find such heuristics. This section will present common approaches to deal with uncertainty. For a more holistic introduction refer to, e.g., Russell & Norvig (2009, p. 610 ff.).

[2] In this domain, the term "risk" is used differently than in the ISO 26262 (ISO, 2011, Part I, p. 13).

5.2.1 Dynamic Bayesian Networks [3]

Bayesian
Networks

Bayesian networks allow probabilistic reasoning based on the idea of conditional probability. Bayesian networks graphically represent relationships between random variables. Every node in a Bayesian network stands for a random variable and directed edges among nodes encode information about the conditional dependence of the random variables. A Bayesian network is a *directed acyclic graph* $\mathcal{G} = \{\mathcal{X}, \mathcal{E}\}$ with a set of n_X nodes $\mathcal{X} = \{X_1, ..., X_{n_X}\}$ and a set of directed edges $\mathcal{E} = \{X_i \rightarrow X_j | X_i, X_j \in \mathcal{X}, i \neq j\}$. All nodes on which a node X_i conditionally depends are called $Dep(X_i)$.[4] Hence, a Bayesian network can be written as:

$$P(X_1, ..., X_{n_X}) = \prod_{i=1}^{n_X} P(X_i | Dep(X_i)) \qquad \forall X_i \in \mathcal{X}$$

Evidence or *measurements* can be incorporated by a joint probability distribution $p(X_1, ..., X_{n_X} | m)$.

Value-
Continuous
Random
Variables

Typically, Bayesian networks assume value-discrete random variables for their nodes. However, they can also be used to handle value-continuous random variables. A value-continuous random variable has an infinite number of possible values. Hence, it is not possible to explicitly state conditional probability tables for the set of edges $\mathcal{E}$. There are two ways to address this issue: On the one hand, a value-continuous random variable can be discretized, while on the other hand a random variable can be described in terms of a particular probability density function, which can be represented by a finite set of parameters (cf. Russell & Norvig, 2009, p. 520). In this thesis, nodes for value-continuous random processes (e.g., a *lane traffic flow velocity*) are transformed into value-discrete, for instance binomial random processes (e.g., *lane change beneficial due to velocity gain*) by a sigmoid function. The cumulative distribution function of a normal distribution is used for this purpose.

Dynamic
Bayesian
Networks

A dynamic Bayesian network extends a regular Bayesian network by the temporal dependency among nodes. Special cases of this general framework are Kalman filters or hidden Markov models. In these, a random variable not only depends on its set $Dep(X_i)$ within a time slice, but also on earlier time slices (cf. Figure 10.9 in section 10.3). For the remainder of this thesis, the Markov property is assumed to hold. That is, a system state only depends on current measurement updates and a *finite* history of previous states. For the remainder of this thesis, only first-order Markov processes are considered, which only depend on current measurement updates and the *last* state $P(X_t | X_{t-1}, m_t)$ and not the events that preceded it. This is a common assumption used for every Kalman or particle filter to simplify calculations. Yet, if the system state vector does not comprise all aspects that describe a system's situation (incompleteness) it will cause a loss of information.

[3]Parts of this subchapter have been pre-published by the author in Ulbrich & Maurer (2015a).

[4]This set is often called the set of parent nodes $Pa(X_i)$. Given the hierarchical structure of abstraction in the Bayesian network in Figure 10.9 in section 10.3, a more abstract node like *Lane change possible left* is indeed a dependent child node to, e.g. *Lane change possible left due to infrastructure situation.* Hence, the author calls this set $Dep(X_i)$.

5.2.2 Model Predictive Control

Reviews of Applications

Dynamic optimization has evolved as a widely used approach for decision making. Initially mainly used in the chemical process industry, it now often is applied in the field of automated driving and driver assistance systems. Refer to Wang (2014, Chapter 2) or Lee (2011) for a literature review. For a more in depth introduction to model predictive control, the reader is referred to, e.g., Camacho et al. (2007). For an overview of non-linear model predictive control refer to Grancharova & Johansen (2012) or Grüne & Pannek (2011).

The essence of such dynamic optimization problems is a dynamic model

$$x(t+1) = g(x(t), u(t)),\ x(0) = x_0 \tag{5.1}$$

to describe the evolution of system state $x(t) \in \mathbb{R}^n$ at time t into a future successor system state $x(t+1)$ at the discrete time $t+1$ which is possibly caused or affected by a control action $u(t) \in \mathbb{R}^m$ (Borrelli et al., 2015, p. ii). The system starts in the initial state x_0. $g(x,u)$ is a possibly non-linear prediction model. The dynamic model is used to find a sequence of best control actions $U_T = [u(t), ..., u(t+T-1)]$ such that the objective function is optimized over time horizon T:

$$min_{U_T} \sum_{t=0}^{T-1} q(x(t), u(t)) + p(x(t+T)) \tag{5.2}$$

$p(x)$ is typically referred to as *terminal costs* and is used to factor in whether the goal was reached or not. $q(x,u)$ is often referred to as *stage costs* for the system taking a particular action in a certain (intermediate) state. The optimization time horizon T can either be finite or infinite. Though, for many real world optimization problems, a finite optimization horizon will be sufficient because achievable model-based prediction horizons are also rather limited.

Receding Horizons

However, most models will only allow reasonable predictions over a limited time horizon. Hence, a common approach is to measure the system state after a certain period of time and to re-solve the dynamic optimization problem iteratively. Consequently, only the first few control actions will be executed. The remaining control actions will be recalculated based on newer measurement data. This feedback of measurement information provides the necessary robustness typical for closed-loop systems (Borrelli et al., 2015, p. ii). This is called receding horizon control.

A standard value-continuous, finite-dimensional optimization problem formulation is given by:

$$\begin{aligned}
&\text{minimize } f(x) \\
&\text{subject to } g_i(x) \leq 0 \ for\ i = 1, ..., p \\
&\qquad\qquad\quad h_j(x) = 0 \ for\ j = 1, ..., q \\
&\qquad\qquad\quad x \in X
\end{aligned}$$

where the cost function $f(x)$, the inequality constraints $g_1(x), \ldots, g_p(x)$, and the equality constraints $h_1(x), \ldots, h_q(x)$ are real-valued functions over the Euclidean vector-space $\mathbb{R}^n \to \mathbb{R}$ for the problem domain $X = \{x \in \mathbb{R}^n\}$.

Types of Problems

Very efficient approaches exist for linear and quadratic optimization problems with convex, differentiable mathematical functions $f(x)$, $g_i(x)$ and $h_j(x)$ in continuous optimization domains. If the optimization domain X is finite, the optimization problem is called *combinatorial* or *discrete*. If the optimization domain is $X \subseteq \mathbb{Z}^n$ the optimization problem is called *integer*. If the optimization domain is a Cartesian product of an integer set and a real Euclidian space, the problem is said to be *mixed-integer*, or *hybrid*.

Non-Linear

The mathematical functions $f(x)$, $g_i(x)$ and $h_j(x)$ can be available in an analytical form or through a black box model. In the latter case they are not explicitly known but can be evaluated by querying the black box for certain values of x. For many solution methods it is necessary to know the gradients $\nabla f(x)$, $\nabla g_i(x)$ and $\nabla h_j(x)$, which are not necessarily easy to obtain for real world optimization problems. If $f(x)$, $g_i(x)$ and $h_j(x)$ are non-linear functions, the problem is called a non-linear model predictive control (NMPC) problem.

Uncertainties

So far, all approaches assumed perfect system state estimates without uncertainties. However, real systems will have prediction and measurement uncertainties. An extension to standard model predictive control approaches exists to factor in stochastic aspects. This field is called stochastic non-linear model predictive control (SNMPC). Applications of stochastic non-linear model predictive control to real world problems are demonstrated, e.g., by Blackmore (2006) or Maciejowski et al. (2007) in the aerospace domain to control the position of unmanned aerial vehicles with strong winds and for air traffic control. Weißel (2009, p. 12 ff.) provides a framework to structure stochastic non-linear model predictive control problems (SNMPC) into (1) open-loop feedback SNMPC, (2) closed-loop feedback SNMPC with perfect state information, and (3) closed-loop feedback SNMPC with imperfect state information. Tactical behavior planning for lane changes unfortunately falls into the last, most complicated category. Chryssanthacopoulos & Kochenderfer (2011) contrast open-loop planning with closed-loop MDP planning. Yet, it does not provide an answer to distinguish "closed-loop feedback SNMPC with imperfect state information" from POMDP planning.

SNMPCs vs. POMDPs

In fact, from the author's point of view and according to the discussion in Weißel (2009, p. 17), stochastic non-linear model predictive control (SNMPC) with "closed-loop feedback and imperfect state information" and partially observable Markov decision processes (POMDPs) are rather different ways to describe the same problem originating from different research fields but are very close in the way problems are solved and described.

5.2.3 Markov Decision Processes and Partial Observability [5]

A Markov decision process is a very general framework to formulate decision problems in situations where an outcome is to some extent random but also partly

[5] This subchapter has been pre-published by the author in Ulbrich & Maurer (2013).

under the control of the decision making instance. It is a time discrete, stochastic control process.

Constituents of a MDP

Given a finite set of states X and a finite set of actions U to be executed, a Markov decision process models the state transition probabilities between states using a transition function or matrix T that represents the transition probability $p(x'|u,x)$ that an action u will lead to a state x' given the system used to be in the state x in the previous time step. A reward matrix or function R specifies the expected, immediate reward of transitioning from a state x into x' by action u. It is used for planning an – to an optimization criterion – optimal sequence of (future) actions $\pi^* = u_0, ..., u_T$ by maximizing the expected, discounted sum of rewards over a potentially infinite planning horizon T:

$$R_T = \sum_{t=0}^{T} \gamma^t r(x_t, u_t) \tag{5.3}$$

Solution Methods

In this equation, γ is a discount factor to weight future rewards against immediate rewards. Typical solution methods for finding an optimal policy are state-space-based approaches such as *value iteration* or action-set-based *policy iteration*. For a detailed explanation, refer to Thrun et al. (2005, Chapter 15).

Partial Observability

A Markov decision process necessitates direct observability of the state x. In real world systems this is often at least partially unobservable. While it is possible to observe visible features like object velocities or accelerations, other aspects of the state space are hidden; for instance the intention of overtaking another vehicle given the relative velocity towards it. A partially observable Markov decision process (POMDP) provides a framework to accommodate partial observability. It should be noted that state estimates in upstream modules like object velocities, lane curvatures, etc., may become observations for more high level POMDPs.

A partially observable Markov decision process is represented by the tuple (X, U, T, R, Z, O, γ) where:

- X is the set of all the environment and system states x_t at time t.
- U is the set of all possible actions u_t at time t.
- T is the $X \times U \times X \rightarrow [0,1]$ transition function, where $T(x', u, x) = p(x'|u,x)$ is the probability of ending in state x' if the agent performs action u in state x.
- R is the $X \times U \rightarrow \mathbb{R}$ reward function, where $r(x,u)$ is the reward obtained by executing action u in state x.
- Z is the set of all measurements or observations z_t at time t.
- O is the $X \times U \times X \rightarrow [0,1]$ observation function, where $O(x', u, z) = p(z|x', u)$ gives the probability of observing z if action u is performed and the resulting state is x'.
- $\gamma \in [0,1)$ is a scalar discount factor to ensure that the reward is finite even if the planning horizon is infinite.

In a partially observable system, a belief over possible system states is derived.

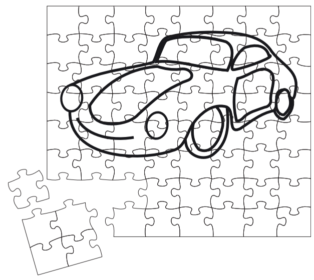
Typically the set of states X, actions U and measurements Z are modeled as *value-discrete*. This increases the computational complexity and therefore partially observable Markov decision processes are often avoided for real-time applications.

The presented implementation will use the separated, non-linear, mixed-integer observation model O, prediction model T, and reward model R of a partially observable Markov decision process to handle uncertainties and non-linearities. It will make use of a finite, receding optimization horizon as in model predictive control to render online solution methods feasible.

5.3 Conclusions

This chapter introduced phases of decision making in behavior planning for humans. In fact, the structure is sufficiently generic to also serve as a basis for technical systems. It will be reused in Chapter 10. Moreover, it laid the foundations regarding probabilistic behavior planning. Dynamic Bayesian networks were introduced to tackle uncertainty; model predictive control and (partially observable) Markov decision processes were presented as general frameworks for behavior planning.

6 Review of Application-Driven Approaches for Tactical Lane Change Behavior Planning [1]



The aim of this chapter is to review application-driven approaches and concepts for tactical lane change behavior planning in automated and assisted driving. The goal is to identify gaps in the existing concepts of tactical lane change behavior planning and its constituting building blocks.

Situation Assessment

Significant research effort has been invested into developing modules for situation assessment of particular driving tasks; of advanced driver assistance systems and automated vehicles as a whole. Section 6.1 reviews application driven approaches for situation assessment for lane changes.

Predicting Driving Behavior

Predicting behavior is central to planning appropriate driving decisions. For this, lane changing can be decomposed: Lane changing is based on longitudinal and lateral driving movements. Many models exist to simulate and predict driving behavior in simulation environments. Many lane change behavior models base their analysis on a situation assessment with a general car-following model applied to the particular situation of the involved vehicles. Thus, longitudinal control models are reviewed in section 6.2.1 as a basis for the subsequent review of lane changing and gap selection models in section 6.2.2. Section 6.2.3 focuses even further on the particular aspect of maneuver and intent prediction.

Overall Behavior Planning

This chapter is completed by reviewing application specific approaches for lane change decision making and overall behavior planning in section 6.3. Here the focus is drawn to model predictive control and (partially observable) Markov decision processes to be used as frameworks for behavior planning.

6.1 Lane Change Situation Assessment

Situation Assessment

Aspects to assess the situation for lane change behavior planning are a key part in the decision making process. While such a situation assessment may be *part* of a general driver behavior model, several more refined situation assessment approaches have been published for certain aspects of a situation assessment. Several methods for the aspect of lane changes have been proposed and implemented and have already been reviewed in Ulbrich & Maurer (2013) and Ulbrich & Maurer (2015a).

[1] Parts of this chapter have been pre-published by the author in Ulbrich & Maurer (2013), Ulbrich & Maurer (2015a), Ulbrich & Maurer (2015b), and Ulbrich et al. (2015f).

Analytic Models

Jula et al. (2000) provide a set of equations to evaluate whether a lane change is possible due to other vehicles around the considered vehicle. They use a model of two sequential phases. First, an adaption of the longitudinal dynamics is assumed. Secondly, the lane change maneuver itself is planned. Their model assumes constant velocities for every surrounding dynamic element. Only the lane changing vehicle is assumed to adopt its accelerations according to a constant value for each phase. Kanaris et al. (2001) chose a similar approach as Jula et al. to calculate a minimum safety spacing for lane changing but also considered more dynamic maneuvers of emergency braking during lane changes. Both approaches were purely analytical and not tested in a real world system.

Fuzzy Logic

Pellkofer (2003) and Naranjo et al. (2008) use a fuzzy logic for modeling lane change decision making problems. The advantage of such a fuzzy logic approach is its simplicity and computational efficiency. To the author, a drawback is that real world implementations tend to be less structured than analytical models.

Bayesian Networks

Schubert et al. (2010) use a Bayesian network for situation assessment and decision making for lane changes. Deceleration to safety time (DST) is utilized as a central criterion for lane change situation assessment. In Schubert & Wanielik (2011), the authors illustrated how to transform value-continuous (measured) state variables into discrete state variables by a discretization of so-called *situation parameters*. A probability density function for a situation parameter is transformed using an unscented transform similar to an unscented Kalman filter. In Schubert (2012), he performs a more in depth practical evaluation of the proposed Bayesian network. He selects some sample sequences of highway driving and illustrates how situation assessments' expected utilities for performing lane changes vary over time. Moreover, Schubert (2012) illustrates the consequences of uncertainties on the ambiguity of the situation over time. In his dissertation, Schubert (2011) provides a simulation-based and an open-loop evaluation based on recoded real world data from a vehicle equipped with sensors.

Li-sheng et al. (2009) calculate a longitudinal acceleration to perform a gap adjustment and calculate a *minimum safety space* based on motion equations to decide whether a lane change is feasible. The authors simulate the model's performance in selected situations in Matlab. Based on this, Roelofsen et al. (2010) perform a stochastic risk assessment for lane changes with a safety risk metric based on a minimal stopping distance and the time gap towards the succeeding vehicle.

Assistance Systems

In the field of driver assistance systems for lane change assistance, Chen (2009) provides an approach for situation assessment based on the point mass motion equations of an object. In his implementation, he focuses on a more heuristic assessment based on time gaps and time-to-collisions. However, in his theory part, he focuses motion equations for objects. By solving these equations for the necessary acceleration/deceleration to safety he provided the framework for the metric used in this thesis.

Habenicht (2012) covers a similar application as in Chen (2009) in his thesis. He developed a Human-Machine-Interface (HMI) for such lane change applications and evaluated a developed lane change assistance system regarding driver stress and safety assessment on a test track.

Kopf (1993) developed abstract models for different aspects of automated and assisted driving. For different situation aspects, he used multiple impact dimensions to form a “situation space”. Depending on the essence of the situation, he suggests to use different situation subspaces if certain dimensions loose relevance for decision making in a particular situation. He develops models for the target velocity of a vehicle, it’s exposure to hazards, obstruction of the ego vehicle and for other vehicles, a driver state model.

Most of these “models” are simple analytic equations to calculate ratios (for instance to calculate how long/often a driver deviates from a headway time) or simple differences (intended target velocity minus delta velocity for infrastructure aspects and traffic aspects). The model for exposure to hazards and obstructions are more sophisticated. His model for hazard assessment is reviewed in section 12.2. To calculate obstruction during lane changes, he uses the motion equation as in Sparmann (1978) to calculate a resulting deceleration as a function of the distance and velocities of two vehicles and a required safety time gap between both.

6.2 Driving Behavior and Situation Prediction

According to sections 2.3 and 8.2, a situation consists of relevant dynamic elements, relevant parts of the scenery, relevant aspects of a self-representation, goals and values as well as relevant function-specific situation aspects. If each of these aspects and their interactions are predicted in the future, the situation can be predicted as a whole. Depending on the time horizon for prediction, it is a simplifying yet versatile assumption that certain elements of a situation do not change or can simply be extrapolated from their recent past.[2]

Behavior Prediction as a Key

However, predicting the *driving behavior* of the ego vehicle as well as that of the dynamic elements around the automated vehicle is essential for situation prediction. The situation prediction for lane changes can be decomposed into longitudinal driving behavior prediction and specialized lane change and gap selection models. Both are reviewed in the following subsections. Moreover, the focus is broadened in section 6.2.3 towards more general maneuver and intent prediction approaches. All three aspects are deeply linked towards cooperation as in section 4.

6.2.1 Longitudinal Driver Behavior Models

Reviews

This section provides a brief review of longitudinal control behavior models. A more in depth review of car following models can be found in Brackstone & McDonald (1999), Toledo (2003), Muller & Zuylen (2006), and Kanagaraj et al. (2013).

GHR-Based Models

The Gazis-Herman-Rothery (GHR) model (Gazis et al., 1961) is, according to, e.g., Brackstone & McDonald (1999) or Muller & Zuylen (2006), among the most well-

[2] For instance, a human driver assumes a highway to continue after a curve even if he or she cannot look sufficiently far ahead. Yet, predictions are inherently wrong. Relying on such incorrect predictions induces risks. In the example above, a rockslide after a curve may cause an automated vehicle to expose itself to a situation it cannot safely handle. Human drivers often unconsciously accepts such risks. This may be partially due to an overconfidence in experience to some extend also part of human life. For a technical system it may not be acceptable by the same extend.

known models for car following and is founded on a stimulus-response-scheme. It is derived from the concurrently found model formulations of Chandler et al. (1958) in the US and Kometani & Sasaki (1958) in Japan. It was further developed by Heyes & Ashworth (1972), Treiterer & Myers (1974) and others. For an in depth review refer to Brackstone & McDonald (1999) or Muller & Zuylen (2006).

Collision Avoidance Models

As cited in Gazis et al. (1961), Kometani and Sasaki proposed in the year 1959 another car following model based on safety distances, resulting in a collision avoidance model. It was further developed by Gipps (1981) and has widely been used for simulation environments. Among the reasons for the popularity of these models may be that they can easily be calibrated as they mostly need easily providable parameters like the maximal applicable braking deceleration.

Human-Oriented Models

Todosiev (1963) and, according to Brackstone & McDonald (1999), Muller & Zuylen (2006), and Erlemann (2007), R. Michaels started in the year 1963 another stream of research for human-oriented models, which focuses on the psycho-physical properties of a human driver. Since the human perception skills of relative velocity and distance changes are limited, they proposed an *action point* model. An in-depth review of follow-up research is found in Brackstone & McDonald (1999) and Erlemann (2007).

Intelligent Driver Models

More recent refinements to longitudinal driving behavior models have been made by Treiber & Helbing (2001) with the widely used "intelligent driver model" which was improved by the consideration of temporary time gap undercuttings in merging situations by Kesting et al. (2010) in their "enhanced intelligent driver model". Shen & Jin (2012) extended the model to incorporate more fluent speedups in stop-and-go traffic situations.

6.2.2 Lane Changing and Gap Selection Behavior Models

Markov Processes for Lane Changes

Rorbech (1974) developed a very general and – to the knowledge of the author – first lane change model for two-lane motorways. He used a stochastic Markov process to model whether a vehicle is on the left or right lane and if the traffic is free-flowing or constrained. He analyzed that the lane change behavior from left to right and, vice versa, is in fact not symmetrical.

Necessary Braking in Gipps' Model

For traffic flow simulation, Gipps (1986) proposed a framework for lane change decision making in sub-urban driving situations (cf. "arterial roads"). He proposed three central questions for lane change decision making: 1.): Is a lane change possible? 2.): Is a lane change necessary? 3.): Is a lane change desirable? He modeled the urgency of a lane change by the aggressiveness of a gap acceptance. The feasibility of a lane change is based on the necessary braking according to Gipps' car-following model (Gipps, 1981).

Game Theory to Model Interaction

Kita (1999) and Kita & Fukuyama (1999) modeled vehicle interactions in merging situations based on game theory. The decision making is based on the simplifying assumption of purely preferring a lower level of risk quantified by a time to collision. In Kita et al. (2002) the model was extended by considering a leading vehicle, which is bounding a lane change gap to the front. However, the model simplifies the reality by ignoring minimal acceptable gap lengths, relative speed differences and distances to lane ends or obstacles, or any kind of induced cooperative behavior.

Gap Adjustment

Hidas (2002) extended the Gipps (1986) model by elaborating strategies to execute future lane changes even if a lane change is currently not possible. He evaluates not only the direct neighbor gap but also other gaps further away. Moreover, he analyzes that field observations show that lane changes are made into gaps, which according to the initial criteria of Gipps (1981) are not feasible at all. He mentions as a strategy, when a lane change is necessary but not feasible, to slow down or even stop. Vice versa, he analyzes that relative speed differences increase by this, and thus lane changes become even more difficult.

Merging Behavior Models

Hidas (2002, p. 366) and Hidas (2005) analyze that merging situations are just a special case of lane changing because a (ramp) lane ends or is blocked. Particularly, but not limited to these situations, he proposes an improved lane change model. If a lane change is necessary but not feasible the merging vehicle will determine the best possible merge acceleration to make the traffic situation more favorable for lane changes. If no gap seems suitable for a lane change it will pick the most behind gap and will thus decelerate. Moreover, he implemented some kind of cooperative behavior for other vehicles to clear a lane for a merging vehicle. As perfect information is available in that simulation environment, other vehicles will not brake abruptly if a vehicle merges closely in front of them, as derived from the car-following model, but will rather decelerate only marginally and will temporarily violate their headway time gap. This prevents traffic flow disturbances and simulates the skill of human drivers to look and plan ahead of the merging vehicle. This behavior is based on the assumption that the merging vehicle will not suddenly perform a hard braking maneuver. Hidas evaluated the performance of his approach by comparing it with the performance of real motorway mergers as given in the US Highway Capacity Manual (TRB, 2000).[3]

J. Wang et al. (2005) chose a similar approach and application as Hidas (2002) and Hidas (2005). He differentiates his models a bit more into an acceleration, gap-selection, gap-acceptance, and a merge model for a merging vehicle as well as a so-called cooperation model for the vehicles on the main road. Similar to Hidas, his scope is purely limited to a simulation environment. Hence, the cooperation model is rather triggered by a binomial random process. Wang et al. use empirically derived acceptable gap-length and gap-clothing-speed parameters. They also model a reaction time for the driver of the merging vehicle as in Toledo (2003). Wang et al.'s merging model does not seem to model that a lane change is not instantaneous and may have to be aborted. All the other limitations discussed for Hidas' model also hold true for Wang et al. Unfortunately, Wang et al. do not benchmark their implementation with the one from Hidas (2002).

Integrated Driver Behavior Model

Toledo et al. (2003) develop an integrated driver behavior modeling framework. A *lane choice model* is used to determine the best possible lane to be in (short-term goal). A *gap acceptance model* is developed to decide if an immediate lane change is possible or not. If a direct lane change is not possible, a short-term plan is executed

[3] Hidas (2002) and Hidas (2005) based their analysis on the numbers from the 3rd edition from 1994.

for adapting the acceleration behavior to be adjusted towards a best possible gap selected by a *gap choice model*. The longitudinal control is integrated into the driver model by an *acceleration model*. Toledo illustrates the advantages of an integrated model, where the lane changing model has influence on the longitudinal control. He mentions that a lane change is not instantaneous but rather spans an amount of time. However, it seems that the abortions of lane changes are not considered in his model. His planning ahead seems to be limited to equations to consider the effect of, e.g., accelerations. It seems that his model is not able to model complex, future interactions between vehicles. Finally, his model is focused on simulated worlds with perfect information and no uncertainty.

Intelligent Driver Model for Lane Changes

Kesting et al. (2007) broaden their intelligent driver model towards lane changes on two lane roads. They compare the acceleration gain obtained from driving within a lane with that obtained from performing a lane change. Other traffic participants are considered by "minimizing [the] overall braking induced by lane changes". A politeness factor is used to trade off egoistic versus altruistic behavior. Their model is able to handle symmetric lane changing rules (e.g., the USA) and provides extensions for asymmetric lane changing rules (e.g., Europe). Their model is extremely sparse regarding the number of parameters. The lane changing part only requires four additional parameters for which numerical values need to be found. Once more, their model is based on perfect uncertainty-free information and models lane changes as instantaneous without a chance to be aborted.

Continuous Lane Changes

Shen & Jin (2012) extend earlier lane change simulation models by modeling a lane change as a continuous process instead of an instantaneous jump from one lane to another. They use an extended car-following model to evaluate gaps. If a lane change is beneficial, it is evaluated by the relative velocity gain from changing to a neighbor lane. H. Wang et al. (2014) improve Shen's model with an actual trajectory formulation in a Frenet frame coordinate system as in Werling et al. (2010) to describe lateral offsets during a lane change. However, maneuver abortions are still not considered.

Limitations

Toledo (2003, p. 44 f.) provides a concise summary of the limitations of driver behavior models. He summarizes that most models are (1) limited to *independent behavior* between multiple agents, (2) assume *instantaneous behavior decisions* that are not based on a plan of actions over a length of time, (3) *reactive* in basing decisions on present or past conditions and not on an anticipated future, and last of all (4) *myopic* by only considering the immediate environment.

Wrap-Up

In contrast to a real world implementation and very recent publications, many simulation models simplify the world by simulating "lane changing [...] as an instantaneous action" (Hidas, 2005, p. 46), (Moridpour et al., 2010), (Rahman et al., 2013). Thus, any problems like aborting an already started maneuver are not considered. Moreover, none of the models reviewed here consider any kind of uncertainty. Hence, all decisions are based on perfect knowledge about the other vehicle's states, intentions and willingness to cooperate.

6.2.3 Maneuver and Intent Prediction

Research on maneuver and intent prediction evolves around the question of predicting a maneuver or intent before it is actually executed. Research is often focused on three central aspects. First, to predict the lane change maneuvers of traffic participants, second to predict turning maneuvers at intersections, and third to predict the driving intentions of the driver in the ego vehicle. It is linked to lane change planning by possibly enhancing situation predictions based on identified maneuvers and intents.

Existing Literature Reviews

Several literature reviews have already been published on maneuver and intent prediction. Regarding the first aspect, Sivaraman et al. (2013) and Freyer (2008, p. 59 ff.) provide reviews. Regarding the second aspect, Shirazi & Morris (2015) and Rössler (2010, p. 7 ff.) provide a review on intersection turning maneuver detections and predictions. For the third aspect, Doshi & Trivedi (2011) provide an extensive review of driver intention detection before an actual maneuver execution and its prediction.

Lawitzky et al. (2013) present a framework for scene prediction by modeling interactions between vehicles. They provide an extensive literature review. Frese (2012) developed a framework for planning cooperative driving maneuvers for automated vehicles. He addresses how to determine cooperative groups and how to modify trajectory planning to cooperate with other vehicles. His evaluations are based on a simulation environment. He assumes a communication channel to communicate vehicle state variables and maneuvers.

Reichel et al. (2010) elaborate on the concept of situation aspects in Pellkofer (2003) and use it to analyze whether the ego vehicle is part of a convoy merging maneuver or if a convoy is about to merge to the ego lane. Reichel (2013) adds the aspect of driver assistance for emergency trajectories. In his situation assessment, he analyzes which areas are about to be occupied by which vehicles and how to adopt the behavior of the ego vehicle.

Gindele et al. (2010) developed a dynamic Bayesian network for estimating the behavior of traffic participants and to predict their future trajectories. In Gindele et al. (2013) the authors trained such a dynamic Bayesian network and used it for behavior prediction at intersections.

Reachable Sets

Althoff (2010) predicts maneuvers by calculating reachable sets for dynamic elements. He models interactions among dynamic elements and between dynamic elements and the scenery. The advantage of his approach is that it can be combined with arbitrary probability representations. He can model discrete aspects like intention changes, passing orders at intersection/merging situations, or maneuver decisions to be taken. This generality comes to the cost of computational complexity. A relatively simple, three vehicle scenario already takes 210 ms to be predicted over a typical planning horizon on a desktop computer (Althoff, 2010, p. 143). An intersection scenario takes up to 1.65 s to predict (Althoff, 2010, p. 146). He uses value-discretization to form non-parametric probability distributions. To the author, a strength of his approach is the generality it provides regarding uncertain environment perception data. Currently, the computational complexity does not allow an application in a scenario-tree based approach because this would require several thousand prediction steps. To the author, it is more helpful to make use of parametric probability representations for value-continuous aspects found in, e.g., Kalman filters within environment perception modules and only use value-discrete probability distributions for truly value-discrete aspects like passing orders, intention changes, or maneuver decisions. This judgment might change if computational resources are one day several magnitudes bigger than they are today.

6.3 Overall Behavior Planning and Decision Making

Focus

In this section, the focus is on the final step of decision making in overall behavior planning. It may use and entail any of the aspects of the previous sections. However, here the focus is rather on overall solutions for behavior planning and maneuver execution.

Thorpe's Team at the CMU

In the driving journal of the "no hands across America" tour it is mentioned that the vehicle performed an "autonomous lane change". Little information is provided regarding the actual implementation. According to Jochem et al. (1995), it seems that the focus of this lane change implementation was on lateral trajectory planning and closed loop stabilization and not on any tactical planning.

Dickmanns' Team in Munich

Jochem et al. (1995, p. 30) credit Professor Dickmanns' group as *the* exception to other research groups at that time having "integrated lane transition functionality [sic!] into their model based lane keeping system" and having tested it *outside* a lab setting. Dickmanns et al. (1994, p. 70) mention that "in the case of an obstacle in the own lane, and a neighboring lane being free of obstacles, a feedforward generic control time history may be called up with a proper set of parameters which is known to steer the vehicle safely into the neighboring lane". They explain that such maneuvers are "achieved by applying simple rules to a data set composed of the relative states of several other objects. [...] This rule based behavior [is] triggered by special events recognized through vision."

Kujawski (1995) explains the situation assessment and decision making in more detail. He groups the "behavior decision" into three aspects: 1) to "bridge over a tracked vehicle in the blind spot between the front and rear cameras", 2) to assess trajectories to decide which maneuvers are safe, and 3) deciding actions and issuing appropriate driving commands. These steps directly relate to the issues approached in sections 10.3.5, 10.3.2, and 10.2 respectively.

Same, but Different

Despite addressing the same problems, the means in this thesis are different. As Kujawski explains that his automated vehicle is fully blind between the front and rear camera's viewing range, his tracking of objects can only be based on very simple motion prediction along lanes or along a vector of movement. In this thesis, at least some radar sensors exist to perform an object existence update. Unfortunately, this raises the issue of imperfect associations (cf. section 10.3.5). Kujawski assesses whether a lane change is possible based on the quotient of the actual distance of an object and a safe distance that should be maintained based on a time gap towards that object, and an absolute minimum distance towards other objects. The minimum of all those quotients is determined and calculated for future situations up to a maneuver duration of 7.5 s with a time step of 0.1 s. This number means that the automated vehicle will not come closer than this minimum multiple of a safe distance towards any object during a maneuver. The author tried to use similar time gap and time to collision based metrics in Ulbrich (2011) and Ulbrich & Maurer (2013), but the metric in section 10.3.2 proved to be more robust and "human-like".

Role of a Human

According to personal communication with Markus Maurer[4], a former team member of that group, the implementation of Kujawski needed a human to command the lane change maneuvers. In the implementation in this thesis, the lane changes are decided by the automated vehicle and are executed without a maneuver confirmation from a human driver. This drastically reduces the tolerance for false maneuver decisions. For that reason, the author still used maneuver confirmations in Ulbrich & Maurer (2013) in more complex urban domains with less sophisticated algorithms but dropped this safety precaution in favor of a better user experience for the *Audi A7 piloted driving concept* demonstration described in this thesis.

Decision Making

In Pellkofer & Dickmanns (2002), the central decision making is described in more detail. The skill and ability monitoring in section 10.3.2 builds upon the concept of "capability nets" from that publication. Pellkofer (2003, p. 86 ff.) also drafted a rule base for lane change situation assessment and behavior planning. However, the actual implementation is limited to regular driving within a lane and simple turn maneuvers at intersections (Pellkofer, 2003, p. 89 f.).

Urban Challenge

Baker & Dolan (2008) use a set of five modules to compose tactical driving behavior. They have a traffic estimator, a distance keeper, a lane selector, a merge planner and a vehicle driver. The lane selector determines a tactical intent, which lane to pick. The merge planner seeks or waits for an appropriate gap to change lanes. A gap assessment is based on a minimal allowable gap length, a velocity-dependent gap length factor, and the distance of the gap. Their approach has been working with uncertain perception data and has been implemented in Boss for the Urban Challenge.

Montemerlo et al. (2008) illustrate an overtaking maneuver but do not provide many details on the implementation. Lane changing is not handled very differently from avoiding objects within a lane. It is performed as a byproduct from trajectory planning. By assuming low success probabilities for lane change maneuvers in their cost functions, these will be executed as early as possible (Montemerlo et al., 2008, p. 283).

Likewise, Team AnnyWAY used a Frenet frame based trajectory planner to decide lane change maneuvers in moving traffic (Werling et al., 2008). In Werling et al. (2011) the abortion of a lane change is addressed and illustrated. Yet, here the focus is on trajectory planning rather than tactical behavior planning.

The team from Virginia Tech (Bacha et al., 2008) as well as the team from TU Braunschweig (Rauskolb et al., 2008) used a behavior-based approach with arbitration between tactical behavior options. Patz et al. (2008) limited their lane change implementation towards a-priori planned lane changes.

[4]Personal communication on September, 8th 2015.

BMW's Lane Change Decision Making

BMW in its ConnectedDrive project focuses on highly automated driving on highways. Ardelt & Waldmann (2011) and Ardelt et al. (2012) describe that they separate longitudinal and lateral control already on the guidance level. A hybrid, deterministic state machine is used to define the superordinate driving behavior and a decision tree is used as a hierarchical decision making process. The superordinate state is determined by traversing the decision tree, depending on the driving goal derived by the situation interpretation and the current feasibility of maneuvers. Bahram et al. (2014) present an approach for general behavior planning with a tree based approach planning three time steps ahead. Their approach is evaluated in a simulation environment. According to Aeberhard et al. (2015), their team seems to aim to bring these algorithms into the real vehicle as part of their ongoing research.

Broggie's Team at Parma

In July 2013, Professor Broggie's team presented the BRAiVE vehicle for automated driving in rural, urban and highway domains (Broggi et al., 2014). Lane changes are mentioned in the paper but no details are provided regarding the technical implementation or the role of a human to supervise or even to command lane changes. The video footage[5] included two merging maneuvers onto a highway. In one of these two merging maneuvers, the automated vehicle seems to detect being overtaken while merging and delays merging. The video does not include any lane changes on the highway or on multilane streets in free traffic flow.

CMU and GM

The Carnegie Mellon University together with General Motors demonstrated a Cadillac SRX[6] driving 33 miles in suburban areas and on highways in 2013.[7] The car managed merging maneuvers without human interaction. However, the chosen track made merging maneuvers rather simple due to the fact that every on-ramp continued into a separate lane. It was only necessary to perform lane changes that could have been planned a-priori; e.g. onto ramps. No lane changes were demonstrated in free traffic flow or with interaction between vehicles.

Daimler

Daimler recently demonstrated their automated driving competence on the Bertha Benz Memorial Route (Ziegler et al., 2014a) and slightly later, in November 2014, in a video driving through California.[8] According to personal communication with a team member,[9] lane changes were neither necessary nor implemented to complete the drive for the Bertha Benz Memorial Route. In the video from California, lane changes can be seen. However, the video is cut in such a way that one cannot distinguish if they are human triggered or automatically executed. Given that they are not highlighted at all in the video it seems likely that they were not the focus of the team. In a patent from 2011, the focus is likewise on human-triggered and human-monitored lane changes (Fritz, 2011). In a video clip from September 2013[10], Daimler presented an E-class prototype vehicle with the capability to perform lane changes. Unfortunately, no technical details are available. However, from the video

[5]https://youtu.be/PiUZ5NCXu-c visited on 05/02/2016.
[6]http://rtml.ece.cmu.edu/Shuster/index.html visited on 05/02/2016.
[7]https://www.youtube.com/watch?v=SxGY4iH5AAc visited on 05/02/2016.
[8]https://download.comsatmedia.com/cms/cms-gomex/videos/preview/mb_141118_sunnyvale_footage.mp4 visited on 05/02/2016.
[9]Personal communication with Julius Ziegler on March, 27th 2014.
[10]http://daimler.cms-gomex.com/editor.php?keywords=autobahnpilot visited on 05/02/2016.

it looks like they are executed automatically. The video is cut in such a way that it is difficult to see how the vehicle performs with smaller gaps in more dense traffic.

On a tactical level, Ziegler et al. (2014b) presented tactical decision making, not for lane changes, but rather to decide the passing order of vehicles. This approach has been implemented in the Bertha Benz Memorial Drive vehicle to plan a passing sequence around obstacles with oncoming traffic. Bender et al. (2015) extended the approach for deciding passing/overtaking sequences in moving traffic. So far, no uncertainty is considered and the evaluation is based on simulations.

Active Lane Change Assist

The new Mercedes E-class model released in 2016 has a so-called Active Lane Change Assist.[11] It is a evolution of the former Active Blind Spot Assist which is now able to plan a trajectory to a neighbor lane if the indicator is activated for two seconds. The blind spot radars are used to monitor the relevant area around the automated vehicle. Yet, as it is an assistance system with a human driver in the loop, the system is not capable of addressing the tactical aspects of behavior planning being core of this thesis. In the E-class, a human driver is still needed to decide if a lane change is beneficial and to monitor its execution.

Delphi

Delphi presented an Audi Q5 driving once across the US in January 2015. In the videos and media coverage lane changes are not mentioned. Thus, the author does not know what has been implemented specifically. Delphi published a concept drawing[12] related to the project and addressing lane changes.[13] Moreover, a video[14] was released which shows a vehicle changing lanes. Whether it is part of the automation and if or how much a human safety driver is involved in those maneuvers remains unclear.

Autonomos

The Autonomos team that grew out of the Freie Universität Berlin has demonstrated the capabilities of their "MadeInGermany" vehicle in Germany and in 2016 also in the US and Mexico. In total 2400 kilometers of automated driving has been performed in urban domains and on highways (Autonomos, 2016). The videos show the execution of lane changes. To the author, it is unknown if they are triggered by the system or a human. Though, in a video footage[15] it seems like they are triggered by the automated vehicle itself once no other vehicles are detected in the sensor viewing range in the neighbor lane. The video footage of the developer visualization monitor[16] suggest that the sensor systems are used to monitor the environment during lane changes. The lane changes have only been demonstrated on road with very sparse traffic. Thus, no narrow merging, gap adjustments, or cooperative behavior were necessary.

At the time of writing, Google/Waymo has provided little technical information regarding their automated vehicle's abilities regarding lane changes. Given that their

[11] http://media.daimler.com/deeplink?cci=2713701 visited on 05/02/2016.

[12] http://delphi.com/media-old/featurestories/automated-driving—scenario-highway-with-lane-change visited on 05/02/2016.

[13] http://investor.delphi.com/investors/press-releases/press-release-details/2015/Delphi-to-Launch-First-Coast-to-Coast-Automated-Drive/default.aspx visited on 05/02/2016.

[14] https://youtu.be/qeJVFavHVJM visited on 05/02/2016.

[15] `http://ftp.imp.fu-berlin.de/pub/autonomos/media/mexico/Autonomos%20Videos/Overtaking%20two%20Trucks.mp4` minute 1:01

[16] Video in Autonomos (2016) minute 1:06.

Google/ Waymo

perception system is based on 360° multi-layer laser scanners on top of the automated vehicles, their vehicles will have a good viewing position and perception power to simplify the 360° object tracking that is essential for lane changes. Their vehicles avoid many challenges such as blind spots at the side of the automated vehicles or perception limitations due to sensors integrated into the bumpers. Thus, it is likely Google's automated vehicles will be able to perform lane changes. And in fact, a reporter test driving the vehicle mentioned lane changes.[17] Likewise, Chris Urmson, head of the driverless car program at Google, presented some videos showing lane changes.[18] Yet, no technical details are known regarding an implementation.

Tesla Motors

Tesla Motors released an "Autopilot" function to its customers in October 2015.[19] It includes a subsystem to execute human triggered lane changes.[20] The vehicle detects whether the neighbor lane is occupied only based on ultrasonic sensors with a 5 m viewing range[21] and executes a lane change trajectory. However, according to Khobi Brooklyn, a Tesla spokesperson, "Tesla is very clear with what we're building, features to assist the driver on the road. [...] Similar to the autopilot function in airplanes, drivers need to maintain control and responsibility of their vehicle [...]" (Berman, 2015). Thus, the system is what other manufacturers would market as assisted or partially automated driving. In particular the "Auto Lane Change" feature with its 5 m viewing range and the need for a human driver to trigger and monitor a lane change is far from truly automated lane changing. Thus, to the author, this is by no means the advent of *the* automated vehicle as it seemed according to some press articles. This impression may rather be caused by – to a certain extent – ambiguous press statements and certain hyperbolism for more lurid headlines by the media itself. Yet, the "Autopilot" feature as a whole and the "Auto Lane Change" feature in particular may still provide one of the best customer experience with minimal technological effort.

Volvo

Volvo is also heavily involved in automated driving and developing technology for it. According to journalists from CNet[22] and the Guardian,[23] their prototype vehicle was at least in 2014 not able to automatically execute lane changes. Though, according to a press video,[24] lane changes seem among the targeted features for 2017.

---

[17] http://www.spiegel.de/auto/aktuell/google-auto-unterwegs-im-selbstfahrenden-auto-a-969532.html visited on 05/02/2016.

[18] https://www.ted.com/talks/chris_urmson_how_a_driverless_car_sees_the_road visited on 05/02/2016.

[19] http://www.teslamotors.com/blog/your-autopilot-has-arrived visited on 05/02/2016.

[20] https://www.teslamotors.com/presskit/autopilot visited on 05/02/2016.

[21] Interview of Tesla's CEO Elon Musk by NVIDIA's CEO Jen-Hsun Huang at GPU Technology Conference 2015 in San Jose, USA: https://www.youtube.com/watch?v=TDm6Snkle70 visited on 05/02/2016. At minute 7:12 of that video recording, Elon Musk states that "the current hardware suite is [consisting of] 360° ultrasonic sensors, that go up to about 5 m, a forward camera, and a forward radar."

[22] http://www.cnet.com/news/a-ride-in-volvos-autonomous-car-how-the-next-step-in-driver-safety-requires-replacing-the-driver/ visited on 05/02/2016.

[23] http://www.theguardian.com/technology/2014/jun/07/driverless-volvo-s60-car-review visited on 05/02/2016.

[24] http://www.volvocars.com/intl/about/our-innovation-brands/intellisafe/intellisafe-autopilot/this-is-autopilot visited on 05/02/2016.

Sivaraman & Trivedi (2014) developed a predictive driver assistance system for recommending lane changes and accelerations/decelerations for lane change preparation. These are derived from a dynamic, probabilistic drivability map, an extension to traditional occupancy grid maps. Dynamic programming is used to calculate a cost optimal acceleration recommendation. The algorithms have been tested in a research vehicle in real traffic. The concept of a dynamic probabilistic drivability map ensures temporal consistency by averaging cell drivabilities over time. The approach works particularly well when the relative velocities between vehicles are small. With high relative velocities as on, e.g., a German highway with no speed limit, drivability cells may be blurred by fast moving vehicles. Moreover, the discretization in $5\,m$ cells may introduce artefacts in very dense traffic. Finally, on very short on-ramps or in weaving areas in highway interchanges maneuver space might be too limited to permit waiting until a steady state of drivability cell values is reached. Despite these concerns, this is one of the very few implementations that have been published and it has proved its feasibility in real world traffic for lane change planning.

UCSD

In the Stadtpilot project at TU Braunschweig, the author developed a first version of the here presented lane change planning algorithms in Ulbrich (2011), Ulbrich & Maurer (2013), and Ulbrich & Maurer (2014). In these publications the state space and observations were converted into a value-discrete representation and a branch and bound tree search has been performed to find a best sequence of actions. Saust et al. (2012) implemented an approaching strategy for traffic lights as another part of tactical driving behavior planning. It was based on a decision tree with a tree search meta heuristic. Wille (2012, p. 110 ff.) developed an offline path optimization approach to address the stabilization level (cf. section 3.2.2). It has been used to generate a smooth, low-jerk reference path as an input for a subsequent controller. The author agrees with Wille (2012) that path planning should be optimized across different (tactical) maneuvers. While stabilization may even be maneuver-independent, the author is not aware of a way to achieve the same for tactical behavior planning.

Stadtpilot

6.3.1 Model Predictive Control-Based Approaches

Mukai & Kawabe (2006) present a lane change assistance system, which uses model predictive control. They model lane change planning as a mixed integer problem and propose multi-parametric programming as a solution method. Proof of the concept is only given in simulation. The ego vehicle is assumed to perform an instantaneous jump from one lane to another. No real perception data or control execution in a real vehicle has been implemented.

F. Wang et al. (2009) calculate a geometric distance based conflict probability and use it in a model predictive control framework for deciding and executing overtaking maneuvers. It has been implemented in some golf carts but according to F. Wang et al. (2009, p. 367), the relative distances towards the overtaken vehicle are not perceived but are rather communicated by Wi-Fi.

Conflict Probabilities

Nilsson et al. (2013) formulate lane change planning as a model predictive control problem and turn it into a linear programming problem with constraints but wit-

Volvo

hout mixed-integer inequalities. Thus, it is easy to solve with the available solvers in real-time. An evaluation is purely simulation-based.

SPARC

Ruf et al. (2014a) and Ruf et al. (2014b) developed a prediction-based behavior planning approach called the "SPARC" framework. Similar to the approach in this thesis, the prediction model is separated from the reward model. A measurement model is not yet necessary because the framework is based and tested on perfect, simulated data. However, input data are allowed to be uncertain. Thus, the approaches in Ruf et al. (2014a) do seem to scale well with real sensor data. So far the SPARC framework is focused on trajectory planning but it would also be able to cover behavior planning aspects. In Ziehn et al. (2015), the authors show the duality between traditional value-continuous solution methods like calculus of variations and solution methods for value-discrete state spaces as in hidden Markov models and illustrate the advantages and shortcomings with their SPARC framework.

Cooperative Groups

Schwarting & Pascheka (2014) propose a system to consider the costs of other vehicles in a cooperative group and tested the approach in a simulation environment and offline with recorded data from an automated vehicle driving on a highway. In Düring & Pascheka (2014) and Pascheka & During (2015) the authors use a model predictive control framework to resolve conflicts using a decentralized planning approach. They assumed perfect information exchange between vehicles and evaluated their approaches in a simulation.

Scenario MPC

Schildbach & Borrelli (2015) present a scenario-based model predictive control approach for a lane change assistance system. Here, the low level trajectory planning is combined with high level tactical decision making. Uncertainty is reflected by a set of traffic scenarios predicted into the future. An evaluation is performed for a single situation. It remains open how the size of the set of future scenarios grows for real world driving.

6.3.2 Markov Decision Process-Based Approaches and Partial Observability [25]

CMU

Wei et al. (2010) use an analytic dynamic environment prediction model for driving and performing lane changes on freeways. Their model focuses on cooperative behavior with the vehicles around the automated vehicle. They used a set of analytic cost functions for decision making. Their approach did not draw particular attention to uncertainties in the sensor data and their evaluation was limited to a simulation environment. Wei et al. (2011) extend this by modeling the task of single-lane automated driving under uncertainty using a point-based Markov decision problem approximation for the underlying partially observable Markov decision process (POMDP).

KIT

Brechtel et al. (2011) demonstrate the usage of Markov decision processes for lane change decision making. Their decision process's state variables are directly based on measured data such as relative distances and velocities towards surrounding vehicles. On the one hand, this helped to ensure decisions were based on physical

[25] Part of this subchapter has been pre-published by the author in Ulbrich & Maurer (2013).

quantities, while on the other hand it made the overall decision process more complex and thus hard to extend to uncertain measurement data. Brechtel et al. (2014) present an approach for automatically finding a best possible discretization of a continuous POMDP problem. Both approaches are only evaluated on simulated data.

Bandyopadhyay et al. (2013) apply mixed observability Markov decision processes for the recognition and appropriate motion planning while considering human agents' intentions. However, they provided a more general framework rather than focusing on lane change decision situations in particular.

Singapore-MIT Alliance

Liu et al. (2015) use online POMDP approaches for decision making at intersections. They follow a similar approach as in Ulbrich & Maurer (2013) by modeling inherent uncertainty and discretizing the state space. They assume vehicle intentions are hidden. Their set of actions is limited to a sparse set of three longitudinal actions of acceleration, deceleration and maintaining the speed. They generalized their model so much that the DESPOT POMDP-solver (Somani et al., 2013) can be applied. Yet, the reason the author of this thesis abandoned the concept of state space discretization prevails: It complicates the prediction model significantly, induces errors, and allows the resulting POMDP problem's complexity to explode.

6.4 Conclusions

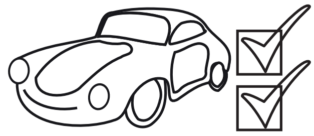
So, what are the key findings in this chapter? This is not the first thesis to address lane changes. Extensive research exists on situation assessment and driving behavior prediction but most implementations are rather conceptual. Most publications ignore uncertainty. While an evaluation based on simulated data is quite common, there are only very few publications on algorithms that have made their way into an actual automated vehicle in real world traffic to solve real world challenges.

From a methodological point of view, there are only a few prior publications that try to leverage model predictive control (MPC) or partially observable Markov decision processes (POMDPs) for tactical behavior planning. Model predictive control-based approaches are commonly found in trajectory planning. Some extensions also exist to reach out to tactical planning. Applying them to tactical behavior planning for lane changes seems promising.

Part II

Concepts and Implementations

7 Requirement Specification for Tactical Lane Change Behavior Planning [1]



This chapter specifies the requirements for an automated vehicle's tactical behavior planning. Schröder (2009, p. 27 ff.) defined a set of requirements for cognitive vehicles: The support of (1) multiple goals and (2) multiple sensors, (3) robustness, (4) extensibility and scalability, (5) the ability to be parameterized to prevailing standards, (6) determinism and traceability, and (7) testability in particular in simulation environments. The first four requirements are based on Brooks (1986, p. 17). Since some of these requirements already relate to an implementation level, the author suggests a set of five meta requirements and rather incorporate the above requirements in subsequent, more detailed functional and useability requirements.

In the author's opinion the requirements for tactical behavior planning are rapidity, consistency, providentness, determinism, and complying with values.

- Rapidity: Behavior planning needs to be fast. Although some strategic decisions (route re-calculation) might be allowed to take some more time, at least most of the tactical decisions for driving need to be made quickly. The longer behavior planning takes, the older the data is that the decision was based on. Due to the dynamic nature of a driving environment, situations change rapidly and hence decisions based on outdated data are supposedly less accurate.

- Consistency: A second criterion for decision making is consistency. In other words, a decision should fit in the framework of the decisions made so far. Similar to a human driver, a decision making unit should not constantly change its mind about the driving maneuvers to be taken. All decisions should align well with a long term goal. However, this does not necessarily imply entirely greedy decision making or not reconsidering a previously made decision at all. For a more detailed definition, see attachment J.

- Providentness: Moreover, behavior planning should have some foresight to predict what the situation will look like after the execution of some maneuvers or simply after some time has passed.

- Determinism: Last of all, behavior planning should be deterministic in the sense that it can be tested and validated according to functional safety requirements.

[1] Part of this chapter has been pre-published by the author in Ulbrich & Maurer (2013).

- Complying with values: Behavior planning should adhere value dimensions to be followed. As in Chapter 9 this entails safety, legality, mobility, user satisfaction, and third party satisfaction.

These meta requirements translate into system requirements. Attachment A provides a more detailed list of system requirements grouped by functional requirements (cf. Table A.1), user interface requirements (cf. Table A.2), usability requirements (cf. Table A.3), and performance requirements (cf. Table A.4). However, due to space restrictions, those requirements are not discussed in detail.

One should note that several requirements in attachment A lack fixed numeric values for pass-fail criteria. Rough estimates could be obtained by educated guesses of an expert group. Yet, a better and more profound way would be to base them on test person studies for those particular aspects. Conducting those studies is beyond the scope of this thesis.

8 Context Modeling for Lane Changes

In a systems perspective, context modeling subsumes any effort to represent necessary information about the environment and the automated vehicle itself. This enables to derive a subsequent – by whatever goal criteria favorable – driving decision. According to Figure 8.1, context information is aggregated from various perception modules into a general context model. Based on this context model, a scene is generated either cyclically or triggered by data events like new incoming sensor data.

Definition

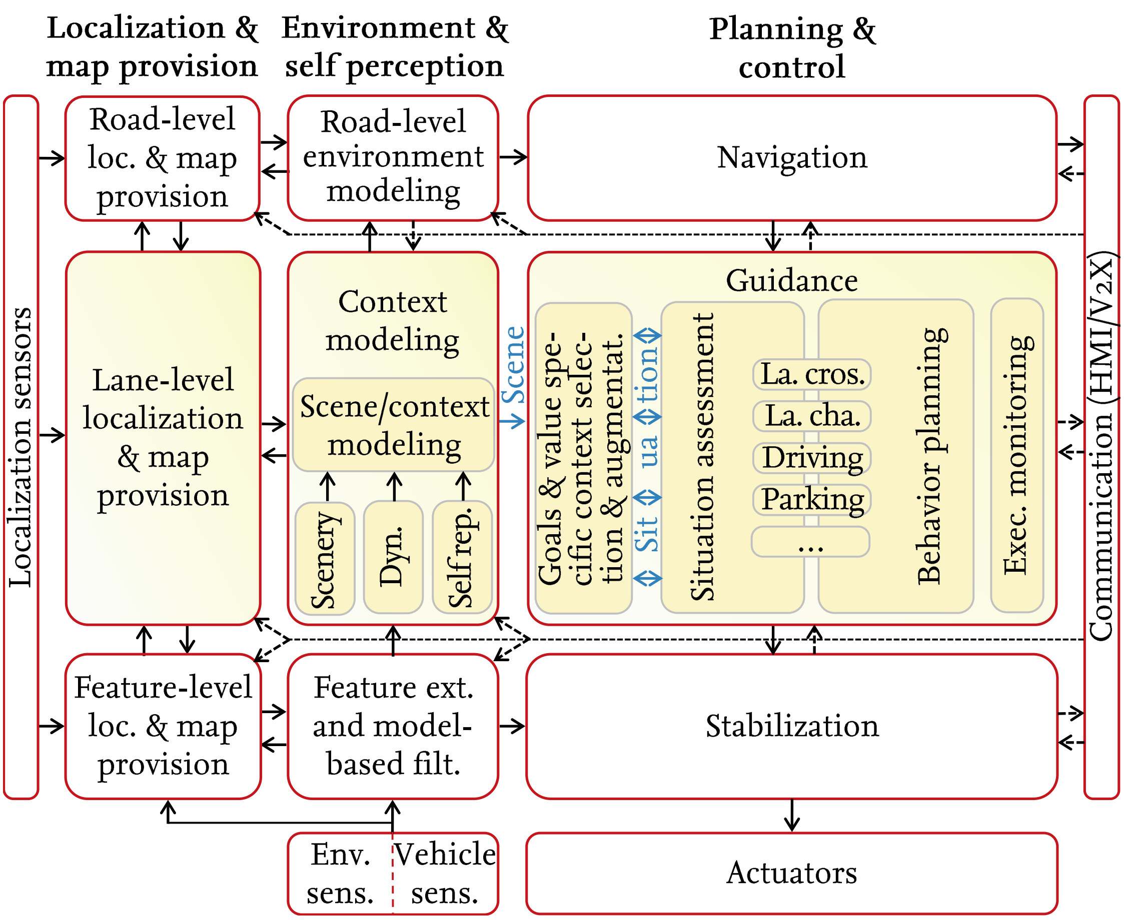


Figure 8.1: A scene and situation in an overall functional system architecture. Modules related to scene or situation interfaces dyed in yellow (scenery = scenery modeling, dyn. = dynamic environment modeling, rep. = representation, augmentat. = augmentation, la. cros. = lane crossing handling, la. cha. = lane change handling, exec. = execution, env. = environment, ext. = extraction, loc. = localization, sens. = sensor, filt. = filtering, HMI = human machine interface, V2X = Vehicle-To-X)

Such a scene may contain irrelevant information for a particular driving task or maneuver. According to section 2.3, the key difference between a scene and a situation is the aspect of an automated vehicle's goals and values. Therefore, decision-relevant context information is extracted from a scene to compose a situation. To extend even to a systemic scope, it may be augmented by decision-relevant goals and values and situation aspects. Figure 8.1 illustrates the process of scene modeling, information selection/augmentation, and a subsequent situation assessment.

8.1 Elements of a Scene Implementation [1]

This section illustrates the scene implementation chosen in the *Audi A7 piloted driving concept* vehicle. It is very similar to the one in the Stadtpilot project at TU Braunschweig (cf. Ulbrich et al. (2014)). Figure 8.2 illustrates the components of a scene. A scene consists of the geo-spatially stationary scenery, dynamic elements, and a self-representation of all actors and observers. Thus, a scene contains information about the environment as well as the system itself (cf. section 2.2).

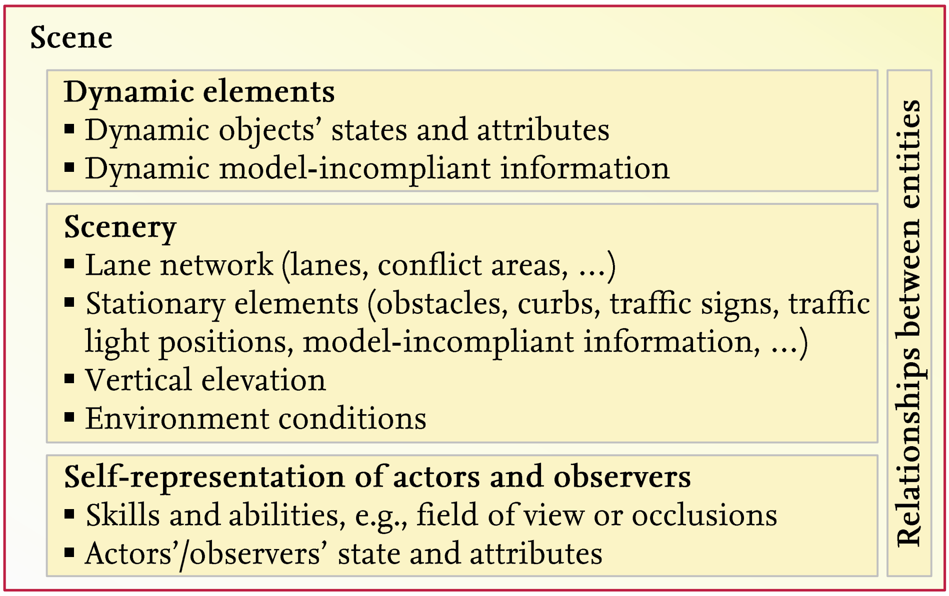


Figure 8.2: Example of a (subjective) scene representation of the real world

Dynamic Elements

Deviating from Geyer et al.'s (2014) definition of "*dynamic elements*" being based on the temporal extent of their scene definition, the author assumes *dynamic elements* to move (having kinetic energy), or possibly being able to move (having sufficient energy and abilities to move). Past movements (object has stopped at traffic lights) are a strong indicator for potential movements in the immediate future. The current perception skills of technical systems are not sufficient to classify stationary elements as *dynamic*; therefore a statue anchored to the ground may currently not be differentiable from a non-moving pedestrian. Hence, a pedestrian may possibly be misclassified as part of the scenery, or a statue as part of the dynamic elements.

[1] Parts of this subchapter have been pre-published by the author in Ulbrich et al. (2015g) and Ulbrich et al. (2015h). The coauthors provided an in-depth review and valuable thoughts on the discussion of coining the term "dynamic elements". Discussions with Jens Rieken caused the authors to include model-incompliant information in Figure 8.2.

Similar to Matthaei (2015, p. 200), the author considers environmental conditions like weather or light to be part of the scenery as they are quasi-stationary for a scene being just a snapshot with an age in terms of milliseconds. Geyer et al. consider the position of traffic lights or variable traffic signs to be part of the scenery, but seem to consider their state as part of the dynamic elements. Based on the snapshot scene definition, the author only requires the scenery to be geo-spatially stationary, thus a changing speed limit sign or traffic light is still considered as part of the scenery.

Traffic Lights and Weather

The *scenery* subsumes all geo-spatially stationary aspects of the scene. This entails metric, semantic, and topological information about roads and all their components such as lanes, lane markings, road surfaces, or the roads' domain types. Moreover, this subsumes information about conflict areas between lanes as well as information about their interconnections, e.g., at intersections. Apart from the aforementioned environmental conditions, the scenery also includes stationary elements like houses, fences, curbs, trees, traffic lights, or traffic signs.

Scenery

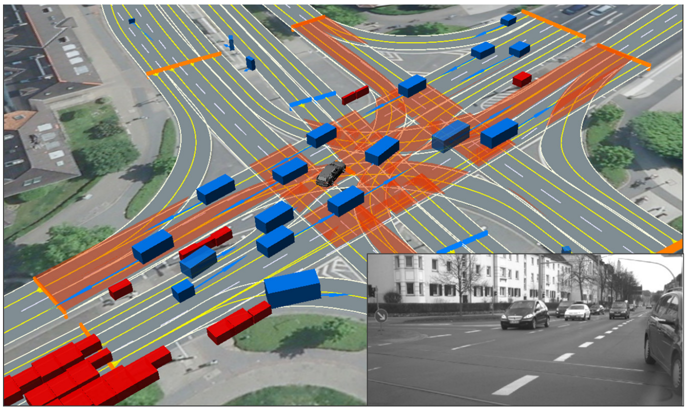


Figure 8.3: Illustration of a (subjective) scene representation from the Stadtpilot project. Image courtesy of Jens Rieken

The scene representation is completed by a *self-representation* containing the current skill levels and general system skills, as well as the states and attributes of all actors and observers. The skills may be represented in a very basic form such as a timeout signal from a sensor system or in the sophisticated form of a skill graph as proposed by Reschka et al. (2015). For observers, the field of view and occlusions are an essential part of its skills. The actors'/observers' states and attributes entail information about the position relative to the road network, dynamic motion information, and even information from the (vehicle's) data busses like whether an indicator is currently activated or not.

Self-Representation

The scene is completed by information that is model-incompliant for dynamic elements or the scenery. This may be unclassifiable, untrackable, or unsegmentable measurements or information about object types not considered at the design time. So far, many implementations simply ignore this information. However, their existence and possibly even partial, imperfect information may be function-relevant from a functional safety point of view.

Model-Incompliant Information

Figure 8.3 illustrates an exemplary screenshot of a subjective scene representation for an automated vehicle with elements and their relationships (e.g., between a dynamic element and a lane).[2] Similar context models or world models with semantic relations have been presented by Vacek (2009), Homeier & Wolf (2011), Ulbrich & Maurer (2014), and Schmidt et al. (2014).

8.2 Elements of a Situation Implementation [3]

Driving Function Relevance

The implementation of a situation deviates from a scene by the aforementioned goal- and value-specific information selection and augmentation. According to Figure 2.1, there is a significant overlap between a scene and a situation regarding the types of information. The major difference is that only driving function relevant information is part of the situation according to the system's goals and values. Figure 8.4 provides an example of an implementation.

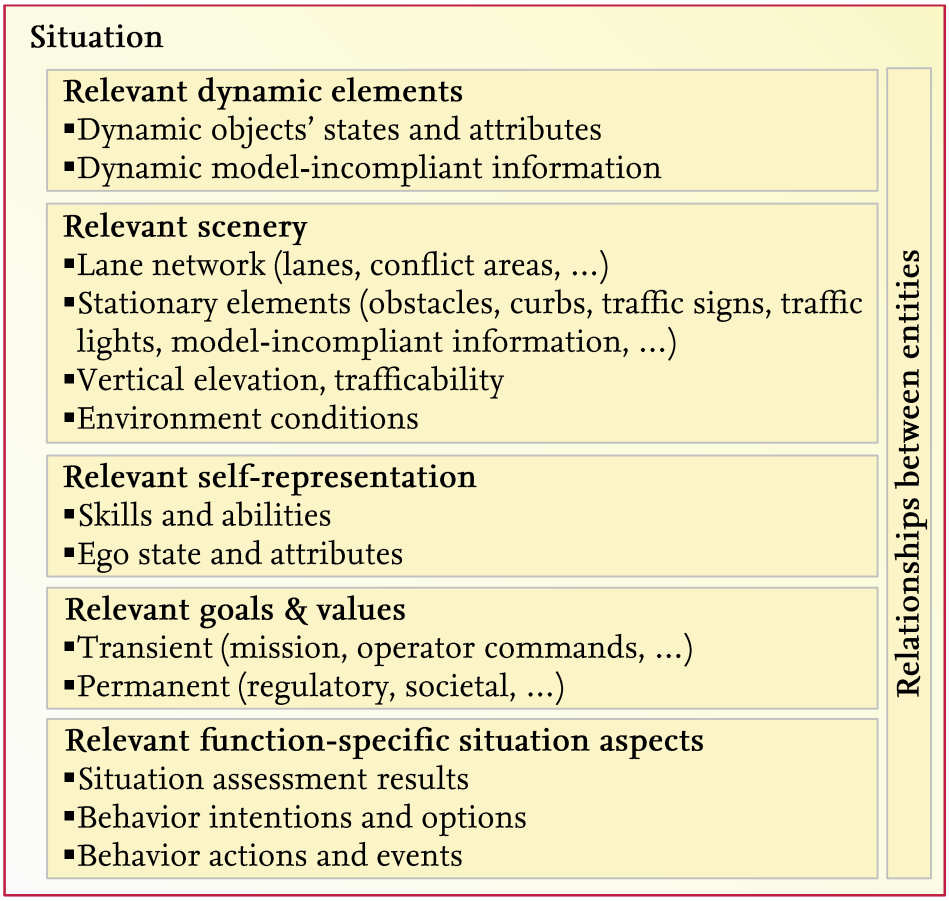


Figure 8.4: Example of a situation representation

[2] The screenshot is taken from the Stadtpilot project, where the author implemented a scene representation in Ulbrich (2011) and Ulbrich & Maurer (2014) together with Tobias Nothdurft (Nothdurft, 2014). Later on, the context model was refined by my colleagues Jens Rieken and Gerrit Bagschik. For the *Audi A7 piloted driving concept* vehicle (cf. attachment B) an implementation was already available. A first concept was published in Homeier & Wolf (2011) and Knaup & Homeier (2010).

[3] Parts of this subchapter have been pre-published by the author in Ulbrich et al. (2015g) and Ulbrich et al. (2015h). The coauthors provided an in-depth review and discussions. Andreas Reschka coined the terms "regulatory" and "societal" for permanent goals and values.

Illustrative Example

A simple example in Figure 8.5 clarifies the difference: An automated vehicle (blue) approaches an intersection with a bike riding on an edificially separated bike lane heading in the same direction. If the mission requires the automated vehicle to pass the intersection straight and the bike has physically no chance to leave its bike lane, it might be irrelevant for the driving function. Thus, the bike would not be part of the situation representation. If the mission requires a right turn and thus a crossing of the bike lane, the same bike is very relevant for the driving function and needs to be part of the situation representation. The scene representation needs to contain the bike at all times as it is independent of the automated vehicle's goals and values.

Goals and Values

Apart from the aspects discussed already in the scene representation, a situation needs to contain function relevant *goals and values*. These may be transient like a current mission, or driving commands and preferences given by an operator to the automated vehicle. In a partial automation, such driving commands may be commanded maneuvers such as lane changes or a changed time gap for longitudinal distance keeping. These goals and values may also be permanent like regulatory or societal constraints. On a systemic scope, they enable decision making according to a country's road traffic regulations or even to accustom to informal rules of behavior in the overall traffic system.

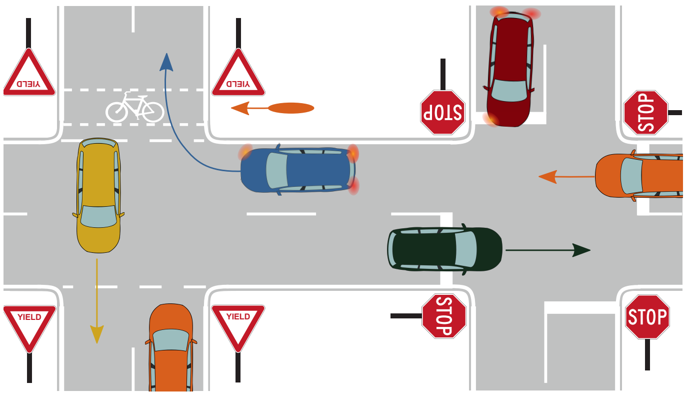


Figure 8.5: Illustration of a situation representation. Automated vehicle in blue.

8.3 Situation Aspects for Lane Change Behavior Planning

In this section the chosen state space is discussed in detail. A situation is used as input for the lane change situation assessment and behavior planning. The situation aspects (Pellkofer, 2003, p. 51 ff.) for lane changes are augmented by results from the situation assessment. State variables may be distinguished by being observable or being hidden as discussed in section 5.2. Observable state variables can be determined directly. Hidden state variables are not directly accessible for inspection and are rather derived by an estimation process.

Observable State Variables

Among the observable state variables is the internal lane change status. The system has perfect information if it has started a lane change, activated the indicator or has aborted a lane change. Thus, the lane change status is observable. Similarly, the time since previous events is relevant for lane change behavior and can be obtained without uncertainty. Among them is the time since the last lane change status change, the time since the last adaptive cruise control following, the time since the last activation of the automated mode, etc.

Moreover, the set of observable state variables entails the currently pursued trajectory as well as a vector of all trajectory candidates of the trajectory planning. Others are the direction of a previously successfully completed lane change maneuver, the object ID of an object which is used for a marginal benefit gain evaluation in a situation where the automated vehicle has to decide to overtake a marginally slower front vehicle (cf. section 10.5.2), and finally the information about whether a lane change was prepared by a gap adjustment (cf. section 10.3.3) before it was executed.

Hidden State Variables

Among the hidden state variables are all the state variables for which no state estimates are available prior to the situation assessment and which are therefore estimated *within* the situation assessment. Among them are situation aspects that are needed for the situation assessment such as traffic flow velocities for the ego lane and the left and right neighbor lane, or a lane change progress estimation from the lateral displacement and a lane width. Moreover, this entails any node in the dynamic Bayesian network (cf. sections 5.2.1 and 10.3). Among them are gap quality estimates for each of the five neighbor gaps around the automated vehicle on the neighbor lane (two to the front, two to the rear, one directly next to it), as well as the situation assessment results about whether a lane change is possible and/or beneficial towards the left and right neighbor lane.

8.4 Conclusions

Scene and Situation

This chapter presented a scene and situation implementation as a foundation for tactical behavior planning. The author defined a generic interface not limited to a specific driving function. Yet the actual implementation may need specific tailoring towards a driving function.

Lane Changes

The situation interface entails application-specific situation aspects. These are detailed further for lane change behavior planning according to how they have been implemented for the *AUDI A7 piloted driving concept* vehicle.

Limitations

The context modeling presented here is still subject of criticism. Firstly, the snapshot concept for a scene and situation is not fully sound: Every element's state variable entails its position, velocities, and further derivatives to describe its trajectory. The more derivatives are included the less it is a pure snapshot.[4] Secondly, another issue is how to incorporate predictions: To the author, a prediction will always be driving-function-specific and should thus rather be a situation prediction than a more complex and more general scene prediction. If a scene prediction is made, it is still open to how to incorporate the behavioral interactions with the ego vehicle and its planned maneuvers. Thirdly, the scene description itself may not be complete yet. At the time of writing, it includes all the author needed and thought of, but for, e.g., driving in shared spaces it may not yet be complete.

[4]This aspect results from a discussion with Professor Dr. Christoph Stiller on September, 29th 2015.

Situation

Relevant dynamic elements

Relevant scenery

Relevant self-representation

Relevant goals & values

Relevant function-specific situation aspects (for lane changes)

- Observable state variables
  - Lane change status
  - Timings
    - Time since last status change
    - Time since last acc following
    - Time since last activation
    - Time since last pressing rear vehicle
    - Time since last driver steering
    - Time since last measurement update
    - Time since last pressing rear vehicle caused lane change
    - Time since last successful lane change
    - Time since last lane change approval
    - Time since last predictive indicator usage left
    - Time since last predictive indicator usage right
    - Time since last marginal benefit gain caused lane change
  - Current trajectory
  - Vector of all evaluated trajectory alternatives
  - Direction of last successful lane change
  - ID of object to cause marginal benefit gain for lane change
  - Lane change preparation was active before start of lane change
  - ...
- Hidden state variables
  - Traffic flow velocities per lane (ego, left neighbor, right neighbor)
  - Gap quality situation assessment (for 5 neighbor gaps to the left/right)
  - Lane change progress
  - Lane change possible situation assessment (left and right)
  - Lane change beneficial situation assessment (left and right)
  - ...

Relationships between entities

Figure 8.6: Chosen situation aspects for lane change behavior planning

9 Value-Oriented Behavior Planning [1]

Moral values are the foundation for human decision making. Given the recent advances in automated driving, it is more and more a reality that a machine gets into the role of making decisions with moral impact. How should these decisions be made, given there is immanent uncertainty? How can these decisions be based on a framework of values for a machine? In fact, it is even unclear what the relevant values are for an automated vehicle.

Problem Statement

When implementing tactical behavior planning, the author identified immanent conflicts of which the resolution has moral implications. Focus of this chapter is to define a value system for an automated vehicle and demonstrate a way to link it to decision making in tactical behavior planning. Herein, uncertainty is a central challenge. Every technical system is subject to system immanent perception, prediction, and execution uncertainty. Even if an automated vehicle with perfect perception and Vehicle-To-X communication would exist, uncertainty would remain: Earthquakes, landslides, sinkholes, or simply a moose crossing a highway can't really be expected at any time. This illustrates that decision making in automated driving is within an open set of possible scenarios. Yet, making decisions has moral implications. Without executing a lane change, a moose crossing the street might not have been hit. Even worse, real world automated vehicles will have imperfect perception and limited Vehicle-To-X communication. This makes decisions with moral implications less clear and less simple than in textbook examples.

Relevance

Value-oriented behavior planning is relevant because automated vehicles with higher levels of automation take over more and more tasks from a human driver. This goes along with more responsibilities and errors may have lethal consequences. While human errors are part of today's reality, errors of machines will see public scrutiny. For a manufacturer of an automated vehicle, they impose a threat for their brand reputation and in court. If driving decisions are deduced from a framework of values that is accepted by society, it provides a foundation to explain why a certain decision was made in favor of one value while possibly violating other values.

Systemic Impact

Value-orientation demonstrates the systemic scope of decisions to be made in behavior planning. Tactical decision have impact on the environment and the environ-

[1] This chapter has been reviewed and improved by Andreas Reschka, Mykel Kochenderfer, and Markus Maurer. The coauthors provided an in-depth review and contributed to several improvements such as the reorganization of the chapter's structure, highlighted aspects like necessity in the law, and together with them the value dimension of third party satisfaction was developed. Moreover, they provided several inputs to clarify the usage of terms towards an audience from a computer science community. They helped to improve the argumentation regarding the application of partially observable Markov decision processes and added the aspect of "micromorts". Furthermore, they helped to streamline the argumentation by highlighting the central role of planning under uncertainty and its implications for value-oriented behavior planning.

ment affects tactical decisions, vice versa (cf. section 2.2). Decisions shall be based on the laws of the system. Those can either be explicit regulatory laws, societal standards or implicit moral values.

Challenge

Considering multiple values in a value system for driving decisions is hard, because values partly contradict each other and need to be balanced in every decision. This balancing has to result in driving decisions, which are not only safe, but also satisfactory for the users and the other traffic participants. On the other hand, the value system needs to be turned into specific algorithms to be executable in real time for online behavior planning and execution.

First attempts have been made by Gerdes & Thornton (2016) and Thornton et al. (2016). The author builds upon their work and extends it towards a more comprehensive framework of values.

Structure

Key contributions of this chapter are: a) A review of concepts for ethical decision making and the definition of a value system for automated vehicles in section 9.1. b) The transformation into the dimensions of utility and risk in a utilitarian framework and a discussion of its implications in section 9.2. c) Pointing out an approach to balance between different objectives in driving decisions in section 9.3. d) Sections 9.4 and 9.5 explain how to substantiate the dimensions of utility and risk towards the lane change problem.

9.1 Values and Ethical Concepts

Mobility, Legality, Safety

The NHTSA accentuated "ethical considerations" to one of eleven "cross-cutting areas" where manufacturers shall comment on for a safety assessment letter to be submitted prior to testing and the deployment of automated vehicles (NHTSA, 2016). Gerdes and Thornton presented[2] value-oriented behavior planning for automated vehicles as a struggle between mobility, safety, and legality (Gerdes & Thornton, 2016; Thornton et al., 2016). To the author, the dimensions of user and third party satisfaction are missing here. Additionally, an itemization into more concrete values is necessary to allow the definition of metrics for values.

Defining Values

In cognitive psychology, dilemma or polylemma[3] decision situations are analyzed for humans. According to Fischer et al. (2007, p. 440 ff.), a human assesses its decisions ultimately on his or her value system. According to Kluckhohn (1965, p. 395) cited by Six (2015), "a value is a conception, explicit or implicit, distinctive of an individual or characteristic of a group, of the desirable which influences the selection from available modes, means, and ends of action."

Roboethics

Some research on value systems for automated systems has already been conducted in robotics and in particular in the sub-discipline of *roboethics*. Roboethics is con-

[2]Keynote speech "Should Automated Vehicles Drive Like Humans or Robots?" at Intelligent Vehicles Symposium 2015.

[3]To the author, a dilemma/polylemma is not necessarily a decision between two/more than two actions resulting in physical damage. Instead, all decisions that include sacrificing one or more objectives for another are dilemmas (two values) or polylemmas (more then two values). Further, there is no investigation of which collision target to choose in a dilemma/polylemma in this thesis.

cerned with the social and ethical implications of robotics (Veruggio & Operto, 2008, p. 1499). An example of an abstract hierarchy of values for an automated system has been defined in the science fiction novel *Runaround* (Asimov, 1942):

Asimov's Laws

- "Law 1: A robot may not injure a human being, or through inaction, allow a human being to come to harm."
- "Law 2: A robot must obey the orders given it by human beings except where such orders would conflict with the first law."
- "Law 3: A robot must protect its own existence as long as such protection does not conflict with the first or second law."

In the novel *Robots and Empires* (Asimov, 1985, chap. 63), Asimov added a zeroth law: "No robot may harm humanity or, through inaction, allow humanity to come to harm." These so-called laws for robots are organized such that obeying a higher-ranked law (e.g., the zeroth) has priority over the subsequent laws (first, second, ...) at any time.

Value System for an Automated Vehicle

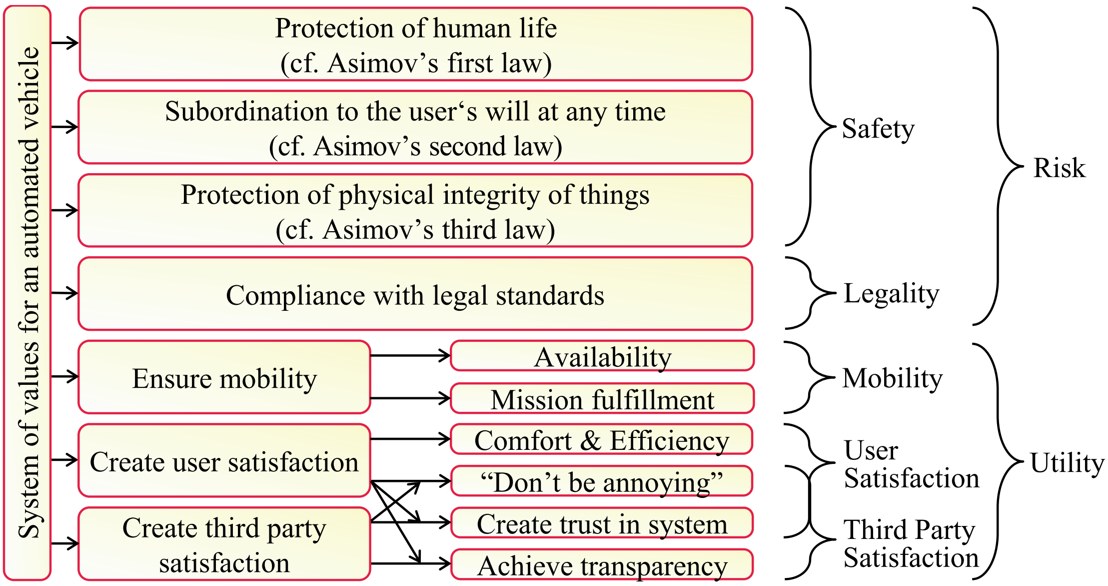


Figure 9.1: Proposed value system for an automated vehicle as an outline of relevant aspects for decision making. Red framed blocks represent values; arrows indicate semantic relationships; brackets indicate summarizing value dimensions

Asimov's laws have been widely criticized due to the slave role they put a robot into (cf. Anderson, 2011). Clarke (2011) provides an overview for discussions about implications and extensions of Asimov's laws for robotics. Asimov provides no guidance in situations where it is not a dilemma between hierarchically different Asimov laws (e.g., killing a person vs. self sacrificing of a machine), but where the essence of the dilemma is that *the situation* makes breaking an Asimov law unavoidable (e.g., deciding between which person to hit, when hitting one of them is physically unavoidable). Another issue is that if Asimov's laws are construed narrowly, their absolutism leave no room for imperfections and uncertainties in perception and prediction. As soon as there is a marginal chance that an object is a human, nothing could be done imposing any negligible risk upon him or her.

Asimov's laws can be used to define a value system for an automated system. Unfortunately, they are quite abstract and it is difficult to convert them directly into program code for an automated vehicle. They also do not give a complete guidance, since the above mentioned dilemma situations are not covered. Thus, Figure 9.1 attempts to outline a more tangible value system for an automated vehicle. The red framed boxes represent values for an automated vehicle. Some of them are comprised of subordinate values. The arrows indicate semantic relationships between values. The brackets relates those towards summarizing value dimensions.

Protecting Human Life and Physical Integrity

As the idea of automated vehicles trying to up-rise against humanity seems beyond the scope of this thesis, Asimov's zeroth law has not been considered. Asimov's first and third laws have been directly translated into the value of *protecting human life* and the *physical integrity of things*.[4]

Obeying Orders

Asimov's second law to *obey orders* has been limited in its scope to just the vehicle's user in order to prevent manipulation or misuse from other traffic participants. Similar to today, where only a driver has control over a vehicle, an automated vehicle should equally only be under the control of its user. Likewise, the group of users may be extended to third parties, e.g., if a vehicle is supposed to be towed away by the road authorities or is under the control of a central traffic operation and coordination unit.

Compliance with Legal Standards

Similar to a human driver, a technical system has to *comply with legal standards*. Thus, this compliance is part of the value system. There might be some situations where a human driver intentionally violates legal standards. For instance, if a parked obstacle can only be passed by crossing a solid line/double yellow line a human driver will do this instead of waiting behind it for hours. Thus, compliance with legal standards may conflict with other values, e.g., the value of mission fulfillment. Yet, it is open whether and how such violations of legal standards will and can be allowed for automated vehicles. If "necessity" as justification applies for automated vehicles as it does for humans, a line crossing is allowed, e.g., to avoid a collision or a near miss.[5] This is similar to German law, especially the German Civil Code BGB Section 904 Necessity.[6]

A question arises about who has to be in control of the automated vehicle, if the user intentionally tries to damage persons or things. In such cases, the control system could prevent the vehicle from crashing into persons or things and thus, avoid harm. An answer to this question cannot be given in this thesis, but it seems necessary to be investigated in the future. Is the machine responsible for minimizing harm although not following the user's will (Asimov's first law over Asimov's second law)? Gerdes & Thornton (2016, chap. 5.6) state that an answer to this question relies on the acceptance of a society "... whether society comes to view these machines as simply more capable cars or robots with their own sense of agency and responsibility" (Gerdes & Thornton, 2016, p. 101).

[4]In this context, the author understands *things* in the legal way: Everything not human is a thing.
[5]Personal communication in 2016 between Stephen S. Wu and Andreas Reschka, who is co-author in (Ulbrich et al., 2017a).
[6]Personal communication in 2016 between Tom M. Gasser and Andreas Reschka, who is co-author in (Ulbrich et al., 2017a).

Another value that is proposed for an automated vehicle is the value to *ensure mobility*. This entails *mission fulfillment* and likewise the *availability* for a driving task. Compared with the first three values it is definitely the one with lowest priority, but it will still be part of the value system: An automated vehicle shall not to cause (avoidable) traffic jams or not to reach the mission goal at all only because it is trying to evade a situation that may potentially become critical. As before, the physical integrity of things has priority over causing a traffic jam. However, after solving a critical situation with, e.g., an emergency braking maneuver, the automated vehicle may still be expected to clear the road and move to the hard shoulder and not block the road longer than necessary.

Mobility and Mission Fulfillment

In addition to the aforementioned values, the author established two more values. To represent the fact that an automated vehicle will have to compete for economic success, it has to *create user satisfaction*. This may be possible in an emotional sphere by creating enthusiasm, but it is also needed on a technical level by achieving trust in the system and transparency regarding its functionality towards the user. Finally, there might even be a certain acceptance for faults as it exists for other systems: E.g., the continued usage of airplanes despite lethal airplane crashes. Similar to a current lane keeping system that requires the driver to keep hold of the steering wheel, an automated vehicle constantly needs to balance user annoyance: E.g., in highly automated driving it needs to balance user annoyance to ensure the user is able to take over the driving responsibility in a timely manner with means of reducing risk by causing slow-downs or accepting temporary risks.

User Satisfaction

Not only the persons inside an automated vehicle are affected by its behavior, but also the other traffic participants, who pursue own goals, need to be considered. The value *third party satisfaction* combines these. Taking into account the cooperation with them is necessary to avoid unsafe actions and to not annoy them as well. Thus, the user satisfaction can be applied to others by implementing predictable, human-like behavior of an automated vehicle.

It is possible that mobility and mission fulfillment ultimately are just sub-properties of creating user satisfaction. However, it is definitely a central aspect as pointed out by Chris Gerdes' tradeoff triangle of mobility, legality, and safety. As it can even conflict with (other) user satisfaction aspects like avoiding annoyance, it is kept separate.

9.2 Utilitarian Behavior Planning and its Limitations

This section discusses the implications of using a utilitarian framework to translate the value system into the dimensions of utility and risk. Baron & Spranca (1997) differentiate between *protected* and *unprotected* values. Tetlock (2000, 2003) uses the terms *sacred*=protected and *secular*=unprotected values interchangeably. Protected values are those "which are not allowed to be traded off no matter what the consequences [are]" (Dehghani et al., 2008, p. 1280). They "block utilitarian motives by evoking deontological moral rules" (Dehghani et al., 2008, p. 1280). The author understands Asimov's zeroth and first law as such protected values as they are *absolute* in their resistance to trade-offs. In particular, they fulfill the property of *quantity*

Protected Values

insensitiveness to the outcome utility and an *agent relative* meaning that the participation of the agent is relevant. Baron & Spranca (1997) suggest two further human related properties for protected values: The *denial of trade-offs by wishful thinking* and even *anger* if people have to think about the violation of a protected value.

One Thought Too Many

In fact, Grau (2011) reflects that a utilitarian approach to decision making may be unacceptable in the presence of *protected values*. Howard (1980) suggests the concept of "micromorts" as a "one in one million chance of death" to quantify utility. Russell & Norvig (2009, p. 616) derives a cost of a micromort of $50.[7] Imagining a situation where a robot has to decide about saving one human or another with different life expectancies and chances of rescue. Here, Grau (2011) discusses the criticism that not only may a calculated utility for each option be *incorrect* causing a wrong decision, but also there is the "offensiveness of the very idea [...] to calculate a utility at all." Grau (2011) reflects on the moral philosopher Bernhard Williams' (Williams, 1981, p. 18) idea of "*one thought too many*." To Grau's interpretation, the "cold utilitarian logic of the robot exposes a dangerously inhuman and thus impoverished moral sense". Following Grau and Williams it would be deeply immoral to calculate utility with micromorts.

Categorical Imperatives for Machines

Powers (2006) differentiates between a utilitarian and a deontological approach. In a deontological approach, actions are judged by the intention of an individual to perform an action and not by the consequences of an action. The good will and a following action are focused, although the consequences might result in damage to persons or things. Powers points out that it is hard to measure "utility over disparate individuals" and whether one can have ever enough information about future consequences (cf. *consequentialism*, Gips, 2011, p. 244 ff.). Hence, he proposes a *Kantian Machine* ruled by a *categorical imperative* that "some actions ought or ought not be performed regardless of how they affect others". He transfers Kant's categorical imperative to a moral agent, which has to "test each maxim (or plan of action) as though it would be a candidate for a universalized rule". A universalized rule has to fit into a system of rules for all agents. The machine should apply this universalization step to individual maxims followed by a mapping onto deontic categories like forbidden, permissible or obligatory actions. In fact, this idea fits well with Mackworth's concept of *constraint fulfillment* for decision making (Mackworth, 2011) and even scales to uncertainty with an implementation by Markov Logic Networks (Richardson & Domingos, 2006). While being founded on a sound theoretical framework, such approaches seem, at the time of writing, to lack computational efficiency and scalability to problems of the complexity of real-world lane change behavior decision making for automated vehicles. Using a Kantian Machine turns an otherwise easy to validate, deterministic cost-based utilitarian solution into a multi-dimensional inference problem. Such a Kantian Machine may potentially decide even lethal actions due to not comprehending the decision situation in its entirety.

Indeed, while the author does not deny the morally difficult implications of assigning costs to the compliance with protected values, it still seems to be a viable engi-

[7] Yet, they highlight inconsistency in human decision making as humans tend to accept one micromort for $50, but will most likely not accept their certain death for 50 million dollar.

neering approach to *turn* an abstract behavior planning problem into a technically solvable one. As a matter of fact, incorrectness, incompleteness, and uncertainty in environment perception currently often results in a lack of the ability to even detect whether an element is a trash can or a human. Hence, moral decisions about whether a collision with one object or another is better from a moral point of view still seem a bit out of reach. Yet, they are relevant and need to be addressed if SAE level three to five automation (cf. section 2.1) shall be archived.

Theory Faces the Real World

9.3 Balancing Utility and Risk

This section illustrates the dilemma of balancing utility and risk[8] in tactical behavior planning for automated driving. Figure 9.2 illustrates tactical maneuver decision making under the constraints of limited technology and the acceptable risk-utility-profile derived from the value system in Figure 9.1. The two axis span the dimensions of utility and risk. The yellow areas indicate different behavior options in different driving situations (at different times). Their fuzzy shape and different size shall indicate that neither their utility nor their risk are fully known in advance.

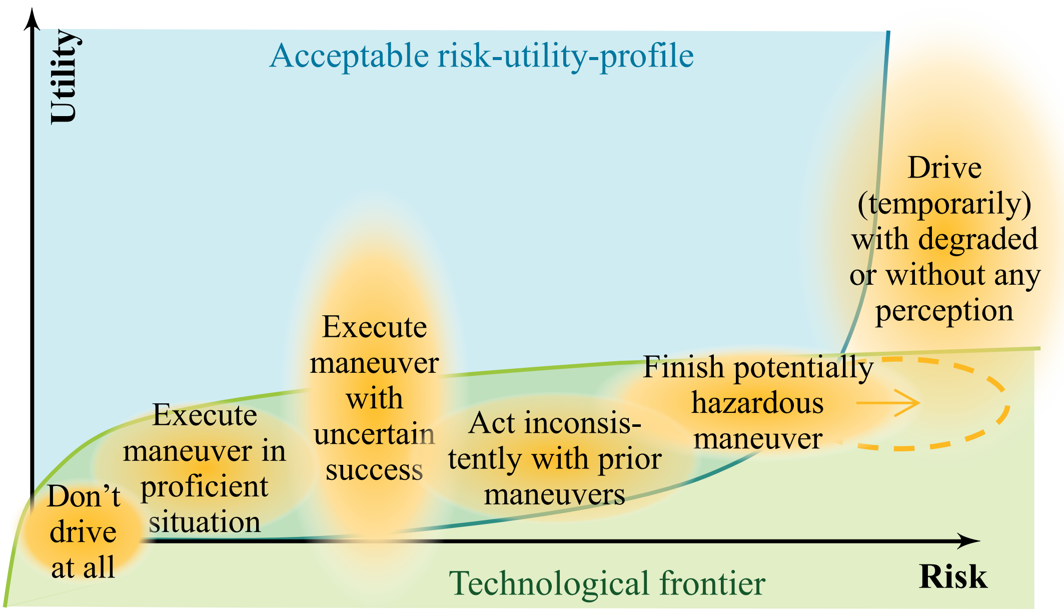


Figure 9.2: Exemplary trade-off between utility and risk for an automated vehicle

The two axes shall serve as aggregated scales for risk and utility. One option for an automated vehicle may always be not to drive (automated) at all. E.g., by not even leaving the garage, because it decides that the weather is too bad for operation. For the case of not leaving the garage, it results in a very low risk (maybe even zero risk),

Risking Utility

[8]Other than in the computer science community, utility and risk are here used as two orthogonal dimensions. Yet, it is still possible to *quantify* utility and risk in a utility function with positive rewards and negative costs.

but also little utility because the commanded mission will not be achieved by the automated vehicle.[9]

Options

At the other end, a possible action might be driving (temporarily) with a degraded or without any perception, e.g., as a result of a technical failure or a sudden change of environmental conditions. Of course, this results in a very high risk of physical harm. Vice versa, it may still be the best remaining option if it is not possible to return control to a safety operator within a few milliseconds or if the potential harm of handing over the driving task to an unprepared, overburdened safety driver is even higher. Whether such an option has a high utility or not highly depends on the situation. This is illustrated by the big vertical stretch of the (uncertainty) ellipse.

As a result, safety is not only about avoiding collisions, but also keeping others and passengers in a safe state, which does not exceed an acceptable level of risk as described in Reschka & Maurer (2015) and Goodall (2014). Risk management is involved in every driving decision, which has to ensure that the remaining risk is always under a threshold and not entirely brought to zero, since this would bring the utility of the vehicle to zero.

Risk-Utility Profile

Given the automated vehicle did leave the garage, it may have to decide whether it shall execute a particular maneuver or not. If the situation is fully mastered by the automated vehicle, this will be within the acceptable risk-utility profile and by definition within the technological frontier. Yet, decision making becomes more complex if the maneuver is no longer fully mastered: There is a certain chance that the maneuver will have to be aborted or may even fail. Lane change behavior decision making with today's state of technology is an obvious example for such a maneuver. It may always be that a lane change needs to be aborted because an object on the neighbor lane has not been detected when the maneuver was initiated. In such a dilemma situation, the automated vehicle has to decide between finishing a potentially hazardous maneuver and acting inconsistently with prior behavior. If the risk of the situation turning critical is still within the acceptable risk-utility-profile (cf. Figure 9.2), it might still be feasible to finish such a maneuver. Of course, it would have been better not to have initiated the maneuver in the first place, but at the given time this is no longer an option. The higher utility, with the still acceptable level of risk, makes finishing the maneuver the better option. Vice versa, if the situation turns worse and the risk of getting into a critical situation increases, as illustrated by the dashed ellipse, finishing the maneuver may no longer be within the acceptable risk-utility profile. Thus, it might be best to accept the user's annoyance and abort an already initiated maneuver.

[9] Counterexamples can be constructed where the inaction of an automated vehicle causes risks. For instance, if an automated vehicle is the only remaining option to transport an injured person to a hospital but service is denied because of bad weather conditions. Here inaction imposes risks and possibly harm.

9.4 Substantiating the Dimension of Risk for Tactical Lane Change Behavior Planning

Defining Risk

In Figure 9.2, the *risk* has been summarized in a single dimension. In psychology, Wirtz (2015b) defines risk as the particular characteristic of a situation, which holds out the prospect of possible harm or losses due to the lack of foreseeability for the upcoming. In the ISO 26262 standard, risk is defined as the "combination of the probability of occurrences of harm [...] and the severity [...] of that harm." (ISO, 2011, Part I, p. 11). However, from the author's point of view this risk definition is not fully sufficient for automated driving. The ISO 26262 limits *harm* exclusively on "physical injury or damage to the health of persons" (ISO, 2011, Part I, p. 9). It would still be unacceptable for an automated vehicle to cause car body damage on a regular basis while avoiding damage to persons. Thus, the author suggests defining harm in a more general way to include "material damage" and "actual or potential ill effects or danger" as in the Oxford Dictionary (Oxford, 2015b). Essentially, risk could be measured in units of *severity-weighted, expected harm per time interval*. In the ISO 26262 this seems to be considered as a scalar value. To avoid weighting, e.g., human life against material damage, measured in money in the risk definition already, it seems useful to represent the risk with a vector rather than a scalar. Figure 9.3 substantiates the dimension of risk for tactical lane change behavior planning. It decomposes the overall risk into the risk caused by non-mastery of the driving situation, the risk caused by an exposition towards non-mastered driving situations, and the risk caused by the potential severity of resulting harm.

Substantiating the Dimension of Risk

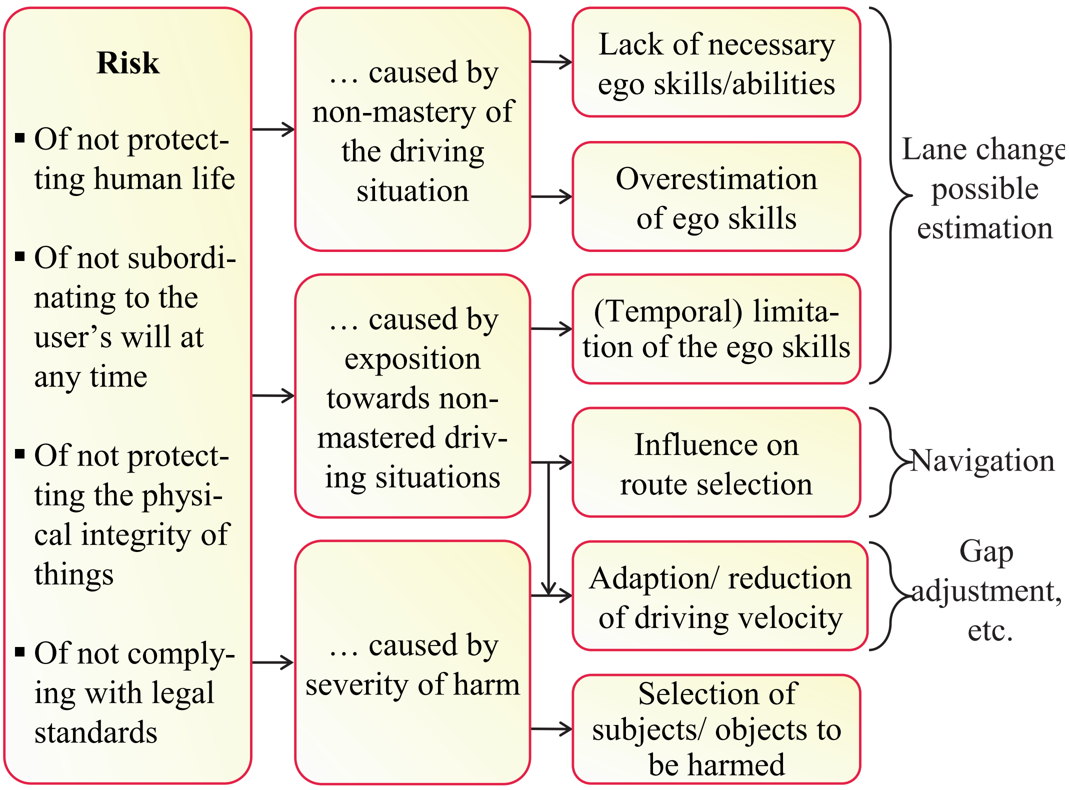


Figure 9.3: Substantiating the dimension of risk for tactical lane change behavior planning

Non-Mastery

The risk caused by non-mastery of the driving situation is ultimately based on the system's lack of skills or abilities. Thus, it can either be due to the lack of necessary ego skills and abilities or the overestimation of ego skills.[10]

Exposition

The risk caused by an exposition towards non-mastered driving situations can be influenced by (temporarily) limiting the ego skills, by influencing the route selection, or by adapting (typically: reducing) the driving velocity. This is performed for gap adjustments (cf. section 10.3.3) or the speed adaption in highway interchanges.

Severity

The risk of not minimizing the severity of harm is not a major concern for this thesis. One day an automated vehicle may be able to decide to hit a big truck rather than a pedestrian to minimize the severity of an unavoidable collision. However, an investigation of decision dilemmas regarding collision targets is not part of this thesis.

9.5 Substantiating the Dimension of Utility for Tactical Lane Change Behavior Planning

Defining Utility

In Figure 9.2, several aspects are aggregated into a single *utility* dimension. According to Merriam-Webster (2015d), utility is "the quality or state of being useful". Thus, it subsumes anything that is favorable to the user or the system as whole. Among those factors is to (1) ensure mobility and mission fulfillment, (2) ensure comfort and efficiency, (3) creating trust, (4) achieving transparency to the user, and (5) avoiding annoyance.

Determine Benefit

The first two items contribute to the functional aspects of lane change planning itself. They mainly relate to a lane change beneficial estimation in tactical behavior planning. According to Figure 9.4, its major three constituents are to determine if a lane change is beneficial due to the infrastructure, if a lane change is beneficial due to the dynamic traffic situation, and if a lane change is beneficial due to timing restrictions. These components will be explained in the description of the measurement model implementation in section 10.3.

Trust and Transparency

The aspects of creating trust in the system and transparency to the user contribute to the dimension of utility. Achieving transparency is related to providing a human-machine-interface that enables the communication of decisions and reasons and ensures mode awareness. After more than one hundred guests in the *Audi A7 piloted driving concept* vehicle, it is very clear to the author that even if a human passenger is not closely following the situation, tactical driving maneuvers have a disruptive influence. In particular, if a passenger is not familiar with the automated vehicle, he or she immediately wonders what the automated vehicle plans to do and why. Thus, it is particularly important for tactical behavior planning to communicate decisions and reasons to humans inside the vehicle. Moreover, it is of great importance to achieve mode awareness. This contributes to both the dimension of utility and risk.

[10] An example for the overestimation of the ego skills is if the automated vehicle estimates a higher viewing range than it is actually able to perceive environment elements. Overestimating ego abilities may only be possible at the time of system design and will thus automatically result in a lack of ego skills.

While it is very dangerous if a driver is not fully aware of his responsibilities during driving, it contributes to the dimension of utility by also creating transparency to the user. Only if a match of the true system skills and abilities and user expectancies is achieved, trust and ultimately customer contentment is accomplished.

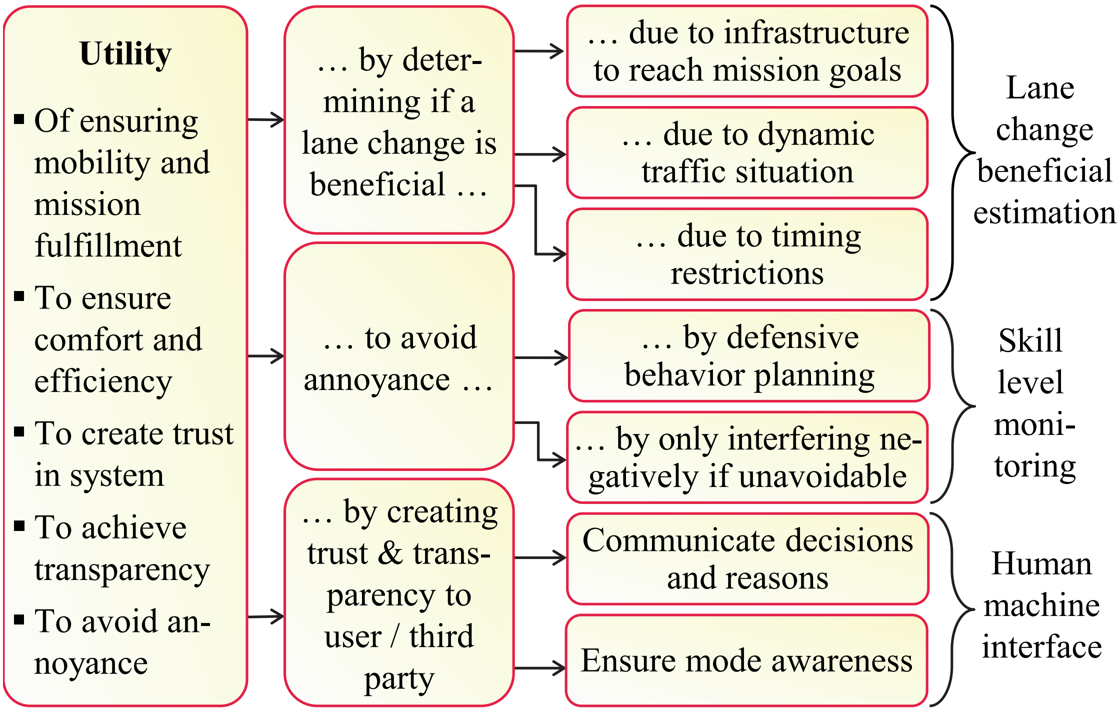


Figure 9.4: Substantiating the dimension of utility for tactical lane change behavior planning

Limiting Trust

The idea of trust and transparency extends both ways: On the one hand to improve trust, if the system is capable of more than a user expects it to do, but also vice versa: It is very dangerous if the created trust in the system's skills and abilities surpasses the true system's skills and abilities. For instance, in scenarios of partial automation, this may result in distraction and even ultimately non-compliance towards what the system expects a user to do (e.g., monitoring) and what the user expects the system to *be able* to do. Therefore, in partial automation, the system may even have to limit trust in its skills and abilities to avoid a mismatch.[11] In higher levels of automation, there should – by design – not be a mismatch between *available* system skills and abilities and *required* skills and abilities.

Avoid Annoyance

Finally, a key issue for the dimension of utility is to avoid annoying a user of the system or other traffic participants and stakeholders. To achieve this, behavior planning needs to be defensive in the way that it only plans maneuvers with a high likelihood of a successful completion. Moreover, the system should only interfere negatively, e.g., by the abortion of a lane change, if the risk of continuing a maneuver is no longer acceptable compared to the discomfort of a maneuver abortion. In the implementation in Chapter 10 will achieve this by two aspects: On the one hand by permanent ability and skill monitoring in a measurement model in section 10.3

[11]This point is based on discussions with Prof. Dr. Thomas Form regarding the importance of creating mistrust rather than trust for a user.

and on the other hand by constantly planning ahead into the future to avoid the initiation of a maneuver if its necessary abortion is already predictable.

9.6 Conclusions

Dimensions

This article addressed value-orientation in tactical behavior planning. Different dimensions such as safety, legality, and mobility have been suggested. Creating user satisfaction and third party satisfaction are additional to those central value dimensions for an automated vehicle. For a simplified discussion, the values are condensed into a two-dimensional consideration of balancing utility and risk.

Protected Values

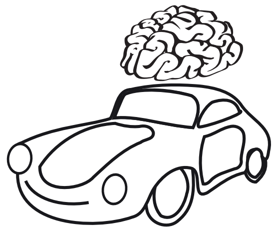
Trading off Utility and Risk

This results in a utilitarian approach. Yet, is that acceptable in the presence of protected values like human life? According to the discussion in this chapter, there are several challenges from an ethical point of view. Even the process of trading off one value against another one may not be moral in a dilemma situation. Yet, it still seems a viable engineering approach to translate an abstract behavior planning problem into a technically solvable one to assign costs to values and decision factors. This turns tactical behavior planning into a constant trade-off between utility and risk.

Substantiating Utility and Risk

Behavior options are limited by an acceptable risk-utility profile and a technological frontier of what is currently possible at the time of the system design. The chapter concludes by substantiating the dimensions of utility and risk towards a technical implementation and results from planning and executing lane changes in a real automated vehicle.

10 Tactical Behavior Planning for Lane Changes



This thesis is dominated by the two major aspects; on the one hand the world of academic models and foundations, on the other hand the challenge to find a working solution for real world driving with incorrect and incomplete information and immanent uncertainty. This chapter presents the implemented approaches for behavior planning under uncertainty. This chapter's focus is on the guidance task to address tactical planning as outlined in Chapter 3.

Pro-Active vs. Re-Active

Chapter 5 introduced different general approaches for decision making to address the trade-off between pro-active, goal driven acting and reactive, rather event-driven (re-)acting. Moreover, section 5.2 introduced the concept of beliefs and uncertain knowledge.

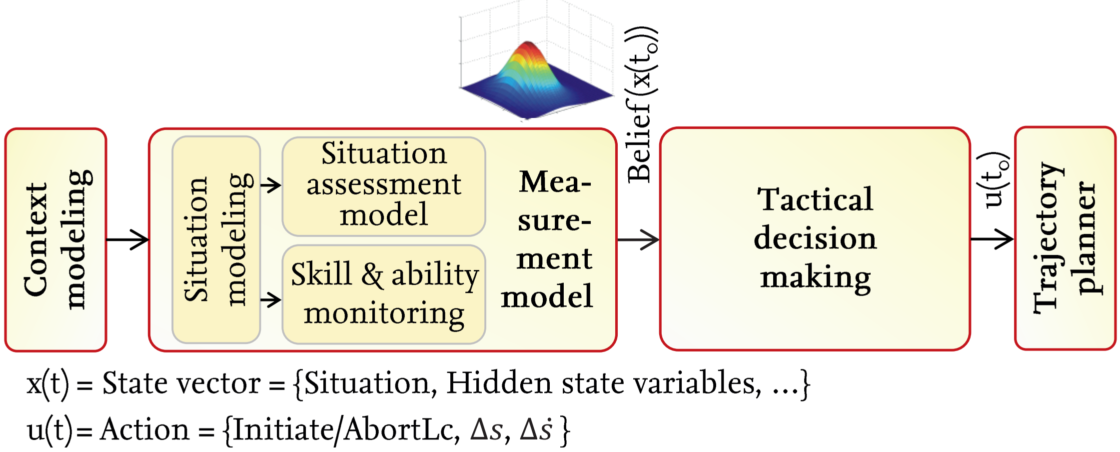


Figure 10.1: Reactive decision making for tactical lane change behavior planning

Measure-ment Model for Situation Aspects

To the author, there should not be any distinct differentiation between pro-active and reactive behavior. Both concepts should rather fade into each other to the extent needed for a particular task. Direct, purely reactive decision making shows a direct mapping $Z \Rightarrow U$ of an observation Z to an action U to be executed. It is fast, but it runs the risk of lacking persistence in reaching its goals, and in some implementations it might even lack the goal orientation itself. It is common to translate an observation Z into some situation aspects, or percepts (Pellkofer, 2003; Siedersberger, 2003; Wooldridge, 1999) P by conducting a situation assessment first. These situation aspects help to compress $Z \Rightarrow P$, a possibly infinite-dimensional observation space Z into a finite dimensional, easier-to-manage, more abstract situation description P. A set of decision rules can be used for a more generalized mapping $P \Rightarrow U$. Furthermore, section 5.2.3 introduced the concept of maintaining an internal state to incorporate some kind of memory in the decision making process. The overall system state (the environment state as well as the internal state) will be denoted as X. Hence, an observation or measurement model provides a mapping $\{Z_t, X_{t-1}\} \Rightarrow X_t$.

Planning Ahead

Figure 10.1 shows an abstract description of reactive decision making for tactical lane change behavior planning. In fact, the procedure in Figure 10.1 does not involve any kind of deliberation or planning ahead. Yet, it is possible to extend it in an effort to achieve that. First of all, a reward model $r(x, u)$ is introduced to quantify the use or harm of a particular action u given the system is in state x. Similar to the belief-desire-intention model (Bratman, 1987; Rao & Georgeff, 1991; Wooldridge, 1999), a planning core is necessary to identify some possible action alternatives called *options* by Wooldridge (1999, p. 29) that are at least potentially good actions to be executed in the given system state x. Each of these action alternatives alter the future system state $x_{t+\tau}$ to some extent. This future system state can be quantified by a situation prediction model $x_{t+\tau} = f(x_t, u_t, \tau)$. The same planning core can be used again to generate (future) decision alternatives for future system states. Those future decision alternatives and states can once again be assessed by the evaluation model. All in all, it is possible to calculate an overall reward over a given planning horizon T as described in equation 5.3. If some aspects of the system state are hidden, the measurement model may be used to derive a belief $b(x_t)$ about the system state based on observations or measurements $z_t \in Z$. Based on this, a reward for belief can be calculated $r(b, u)$ and a future system state belief can be calculated from a situation prediction $b_{t+\tau} = f(b_t, z_t, u_t, \tau)$.

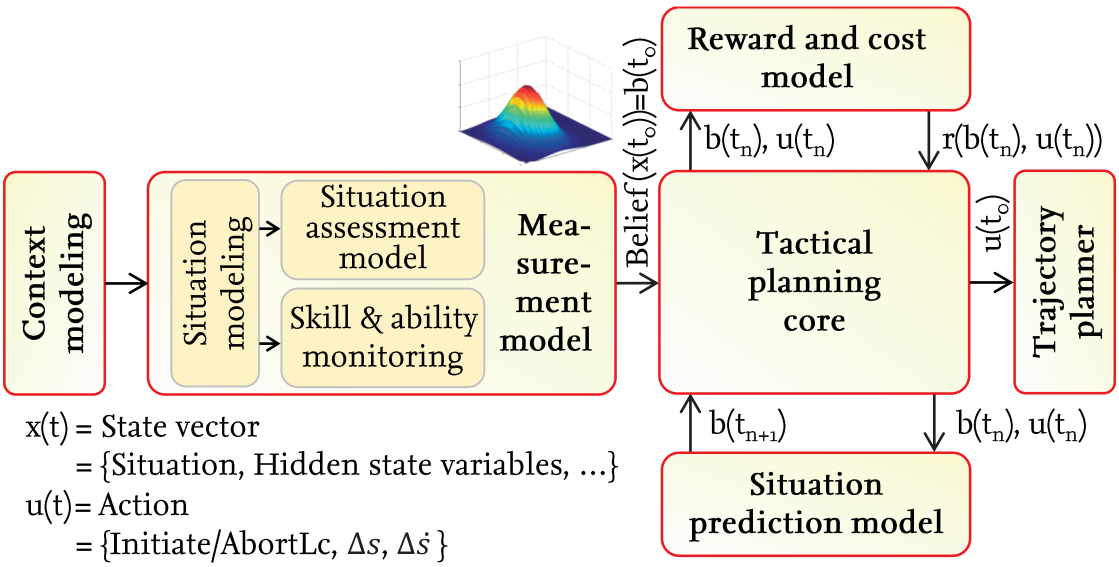


Figure 10.2: Behavior planning: iteratively planning ahead

Figure 10.2 illustrates the expansion of reactive decision making towards a deliberative approach. In fact, if the planning horizon T is set to zero and the action with the highest reward $r(b(t), u(t))$ is executed, Figure 10.2 directly collapses into the process described in Figure 10.1.

10.1 Foundations in Model Predictive Control and Partially Observable Decision Processes

In what way is the behavior planning in Figure 10.2 different from model predictive control and its extensions introduced in section 5.2.2? Behavior planning for lane changes has value-discrete aspects required to be part of the state space. Hence, it cannot fully be represented by linear models as in traditional model predictive control. Additionally, not all aspects are fully observable. Therefore, approaches for stochastic non-linear model predictive control are necessary. Weißel (2009, p. 12 ff.) structures stochastic non-linear model predictive control problems (SNMPC) into (1) open-loop feedback SNMPC, (2) closed-loop feedback SNMPC with perfect state information, and (3) closed-loop feedback SNMPC with imperfect state information. Closed-loop feedback SNMPC with imperfect state information applies to behavior planning for lane changes. As mentioned in section 5.2.2, the author of this thesis shares Weißel's (2009, p. 17) point of view that stochastic non-linear model predictive control (SNMPC) and partially observable Markov decision processes (POMDPs) are rather different ways to describe the same problem originating from different research fields. Thus, they are very close in the way problems are solved and described. To the author, the concepts of deterministic state prediction models, finite horizon planning and multi-resolution planning from the model predictive control community are particularly useful to be applied for lane change behavior planning. Additionally, the idea of constrains will be reflected in the concept of pruning away solutions that are known to be inefficient or infeasible. Vice versa, several of the traditional mathematical solution approaches for model predictive control are not applicable because they are based on the idea of linear models and differentiable functions. If linear approximations do not fit non-linear aspects like singularities in a reward function or in a state transition model well, the concept of gradient descent approaches to find a solution close to a global optimum provides limited use. If mixed integer optimization approaches are employed, the line between SNMPC solution approaches and (online) POMDP solution methods vanishes.

Convergence Towards MPCs

In what way is the behavior planning in Figure 10.2 different from the POMDP decision making introduced in section 5.2.3? Indeed, it is not different at all if the state set X, observation set Z, and action set U are allowed to be infinite-dimensional and are a mixture of value-discrete and value-continuous variables. Yet, as outlined in section 5.2.3, this class of problems renders many solution approximation techniques intractable. Figure 10.3 shows a simplified view on assumptions for state spaces and observation spaces. It distinguishes between value-discrete, value-continuous and mixed state and observation spaces. The simplest form of (stochastic) model predictive control assumes value-continuous state and observation spaces and linear models to describe their changes and relationships. The simplest form of partially observable Markov decision processes assume value-discrete state and observation spaces. Extensions have been made to expand into the other fields as in stochastic non-linear model predictive control or mixed-integer and value-continuous POMDPs. To turn a problem with value-continuous or mixed state and observation spaces into a discrete one, *discretization* is often applied as

Convergence Towards POMDPs and MPCs

an approximate solution method for POMDPs. It is assumed that for instance a principal component analysis can be used to distinguish few decision relevant subdimensions and that sparse discretization patterns can be found in the frontier zone around decision boundaries.

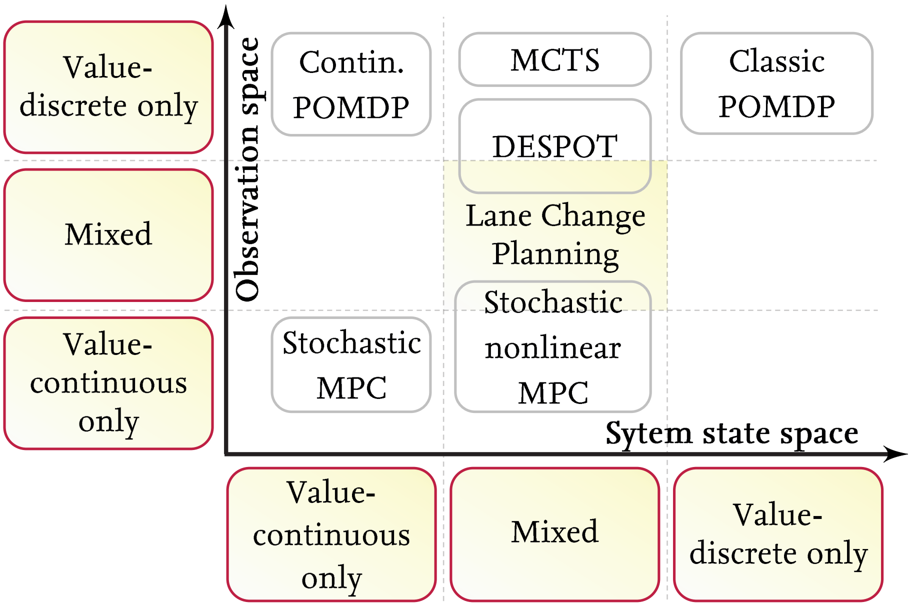


Figure 10.3: Methods for value-continuous and value-discrete state and observation spaces (contin. = continous, POMDP = Partially observable markov decision process, MCTS = Monte carlo tree search, DESPOT = Determinized sparse partially observable trees, MPC = Model predictive control)

Challenge

In fact, the true challenge is then to actually be able to make use of such a simplified state representation for situation prediction. Since situation prediction models in section 6.2 necessitate value-continuous state representations of the situation, it poses a challenge for the overall approach. So, where are typical POMDP solution methods good to be used for? They work well in domains where discrete actions have distinctive, discrete outcomes, where transition functions for probability distributions over the state space can easily be specified, e.g., in a state transition matrix with quantitative transition probabilities. In fact, none of these assumptions hold true for lane change behavior planning. As demonstrated by Ulbrich (2011) and Ulbrich & Maurer (2013), it is possible to enforce a value-discrete model onto those problems, as well. Yet, from the author's point of view, the approach does not really play out its benefits. This is because the discretization is subject to the "curse of dimensionality" (Bellman, 1957, Preface, p. 9) because discretizing an, e.g., a 40-dimensional value-continuous state space into a relatively coarse 10-step discretization pattern for every state variable yields 10^{40} different elements and renders it intractable for fast online computations. Most computational resources are used towards handling the self-induced complexity resulting from the discretization.

Offline vs. Online

Many of the so far published solution methods[1] for POMDPs are executed *offline* to calculate best policies (sequence of actions) for *all possible belief states* in advance (po-

[1]For instance, point-based approximation techniques surveyed in Shani et al. (2013), point-based value iteration (Pineau et al., 2003), the "Successive Approximations of the Reachable Space under

licy generation). Yet, driving in dynamic environments necessitates online decision making being based on new sensor readings several times per second. Thus, decision making is then implemented (policy execution) by applying those previously found best policies by a simple look-up table, decision tree, neuronal network, or k-nearest-neighbor search in order to find at least the most similar states for which an approximately optimal policy had been computed. The challenging and still insufficiently solved task is to find an appropriate distance metric to determine other "similar" states in lane change planning. Vice versa, traditional (stochastic) model predictive control typically assumes *online* solution methods that try to find a best policy not for all possible beliefs but only for a *current belief* by evaluating future beliefs that can be reached from the current belief. The same idea has been transformed towards POMDP approaches in online POMDP solution techniques (Ross et al., 2008). If policy generation is performed *offline* for mixed and value continuous state spaces, it is a challenge to represent those policies. If discretization is applied on high-dimensional value-continuous state spaces, the number of resulting discrete states may even be intractable for the most capable solution methods known today, because of the exponential number of future belief distributions that must be considered.

Searching Belief Trees

For online solution methods it is common to construct a belief tree over a finite planning horizon T. Its root node is the current system state belief $b(x(t_0)) = b(t_0) = b_0$. It branches by all possible actions $u^j_{t_n} \in U$ and all observations $z^k_{t_n} \in Z$. Figure 10.4 illustrates such a belief tree. Different actions may result in different observations with different subsequent beliefs in a future time slice. Once again, in these new future beliefs it is possible to select different actions resulting in different observations. The tree is expanded until the planning horizon T is reached. The path with the highest reward may be selected as a best policy (sequence of actions).

Different solution methods have been presented to find such best policies (cf. Kochenderfer, 2015, p. 149 ff. and Ross et al., 2008). These entail forward search in the belief tree, branch-and-bound pruning, and Monte Carlo sampling based tree search. Monte Carlo tree search has been surveyed, e.g., by Browne et al. (2012) and been specifically applied to online POMDPs by Silver & Veness (2010) and Lenz et al. (2016). It is a probabilistic heuristic to expand the tree of future beliefs, actions, and observations.

DESPOT

A recent approach to online POMDP solving has been "Determinized Sparse Partially Observable Trees" (DESPOT) presented by Somani et al. (2013), Luo et al. (2016), and Ye et al. (2017). Similar as in Monte Carlo tree search the foundation for finding a best policy is a belief tree. While a complete belief tree has a complexity of $O(|U|^T|Z|^T)$, a DESPOT only considers a set of K scenarios sampled from a initial belief distribution b. A scenario not only consists of a state sample s_0 but likewise a set of real numbers ϕ_i sampled independently and uniformly from the range $[0,1]$: $\{x_0, \phi_1, \phi_2, ...\}$. A "deterministic simulative model" $(x', z') = g(x, u, \phi_t)$ is used to simulate a future state and observation according to $p(x', z'|x, u) = T(x, u, x')O(x', u, z')$. If a scenario $\{x_0, \phi_1, \phi_2, ...\}$ is simulated

Optimal Policies" (SARSOP) in Kurniawati et al. (2008), the Perseus algorithm by Spaan & Vlassis (2005).

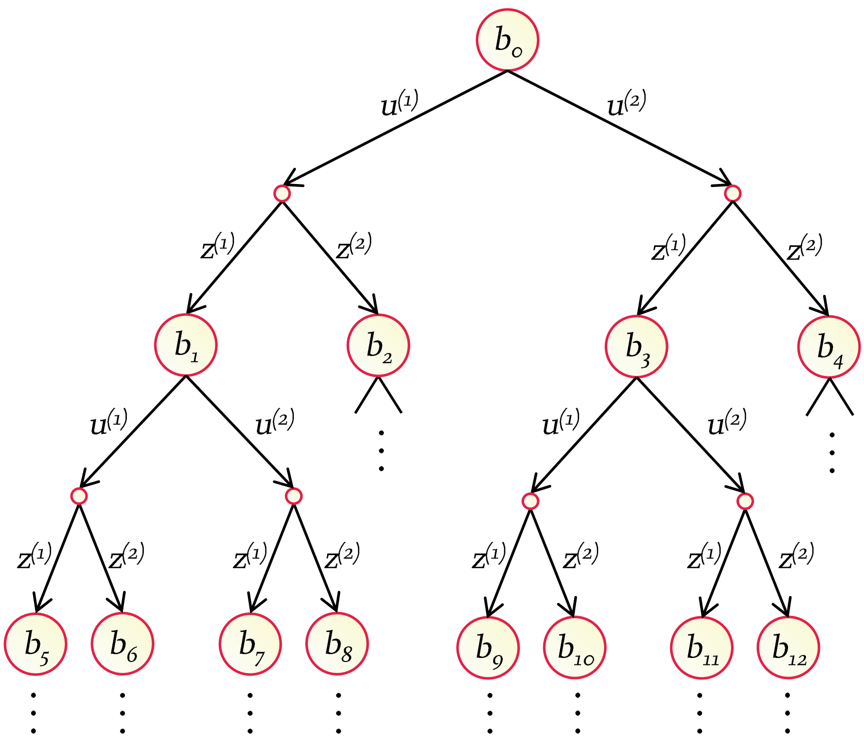


Figure 10.4: Belief tree for decision situation with only two actions $u^{(j)}$ and two observations $z^{(k)}$. Each belief tree node represents a belief. Each path represents an action-observation history.

for a action sequence $\{u_t, u_{t+1}, u_{t+2}, ...\}$ it results in a path in a standard belief tree of $\{u_t, z_t, u_{t+1}, z_{t+1}, ...\}$ from an initial belief node b to a fringe node at the planning depth T. A DESPOT for a scenario contains all these paths and nodes. Each DESPOT node b has a set of all scenarios Φ_b. The scenario $\{x_0, \phi_1, \phi_2, ...\}$ is added to Φ_{b_0} for the tree's root node b_0. The K scenarios form a particle set in Φ_{b_0} to represent b_0 approximately. Accordingly, the sets Φ_{b_t} are formed from $\{x_t, \phi_{t+1}, \phi_{t+2}, ...\}$. Hence, a DESPOT has only $O(|U|^T K)$ nodes, because of the reduced observation branching under the sampled scenarios. The DESPOT approach is extended by a "regularization" step to balance exploration and exploitation in action selection and an "anytime approach" using branch-and-bound[2] to provide a best possible solution at any time of the evaluation.

Application towards Lane Changes

Similar as in a DESPOT approach (Somani et al., 2013), a "deterministic simulative model" will be applied here. The big advantage of a DESPOT over a traditional Monte Carlo belief tree search is that the complexity $O(|U|^T K)$ does not scale with $|Z|^T$. As there is no branching for discrete observations, this approach works well even if the observation space Z is value-continuous. This is particularly helpful for the application domain of lane change planning in automated driving. Here the observation space is value-continuous. Discretizing those continuous observations would only increase the complexity of the problem to solve. Both Monte Carlo be-

[2]The author applied branch-and-bound in Ulbrich & Maurer (2013). Yet, it is very challenging to come up with tight bounds in lane change behavior planning. Thus, branch-and-bound is limited in its usefulness and will not be followed upon in this thesis.

lief tree search and DESPOT are based on a Monte Carlo sampling of a belief. This works well on low dimensional state spaces. The more dimensions the state space has, the more particles are necessary to approximate the probability distribution of the belief. Section 10.2.1 describe relevant state space dimensions based on sections 8.2 and 8.3. This illustrates that the state space for lane change planning is indeed very high dimensional with several value-continuous dimensions. Decreasing the number of state dimensions reduces the quality of the situation prediction. E.g. ignoring the effect of a second front vehicle's acceleration towards the direct front vehicle may cause a significant error if this pursues a strong braking. Even with several thousand particles, the author was not able to match the probability distributions in the state belief sufficiently accurate. Vice versa, significant effort was spent in the model-based filtering modules in the perception part of the overall architecture (cf. Chapter 3) to find sufficiently good parametric probability distributions for every aspect of the scene or situation. Thus, it stands to mind to make use of these parametric probability distributions like the mean or variance of the normal distribution or the parameters of a beta or Bernoulli distribution. Hence, a DESPOT-like approach could be used only without the Monte Carlo sampling but using the parametric probability distributions instead.

What is left?

All in all, what is left from the POMDP and where does it provide value? First of all, the structure of a POMDP as a whole, which separates the overall behavior into a measurement model to obtain state estimates for observations, a prediction model, and a reward model. It provides a clear structure and separates the overall problem into clearly distinguishable sub-modules. Existing research on estimating relevant state aspects can be used in the measurement model and research efforts on situation prediction can be leveraged on for the situation prediction model. Moreover, it is possible to leverage on the idea of belief tree search. Details are provided in section 10.2.

10.2 Planning Core [3]

The planning core uses the state beliefs being updated from the measurement model in order to derive behavior decisions. It utilizes a reward and situation prediction model as illustrated in Figure 10.2. In the end it decides the best action to pursue and hands it over to the consecutive blocks for trajectory planning.

10.2.1 Mutually Exclusive, Collectively Exhaustive States or Actions

Given the measurement model and the overall concept for behavior planning, what does the state space for lane change planning actually look like? Lane change planning uses the entire situation representation as in sections 8.2 and 8.3 as a state vector. This entails among others the ego velocity and acceleration, the front vehicles (positions, velocities, accelerations, existences), the left/right neighbor lane's front vehicles (positions, velocities, accelerations, existences), the left/right neighbor lane's rear vehicles (positions, velocities, accelerations, existences), the course,

[3] Parts of this subchapter have been pre-published by the author in Ulbrich & Maurer (2015b) and Ulbrich & Maurer (2013).

drivability, existence, and usefulness to reach the navigation goal, of the ego lane, and left/right neighbor lane. The situation aspects for lane change planning from section 8.3 entail a "lane change status". This is a value-discrete state variable forming mutually exclusive, collectively exhaustive lane change states that are central for tactical behavior planning. They are illustrated in Figure 10.5.

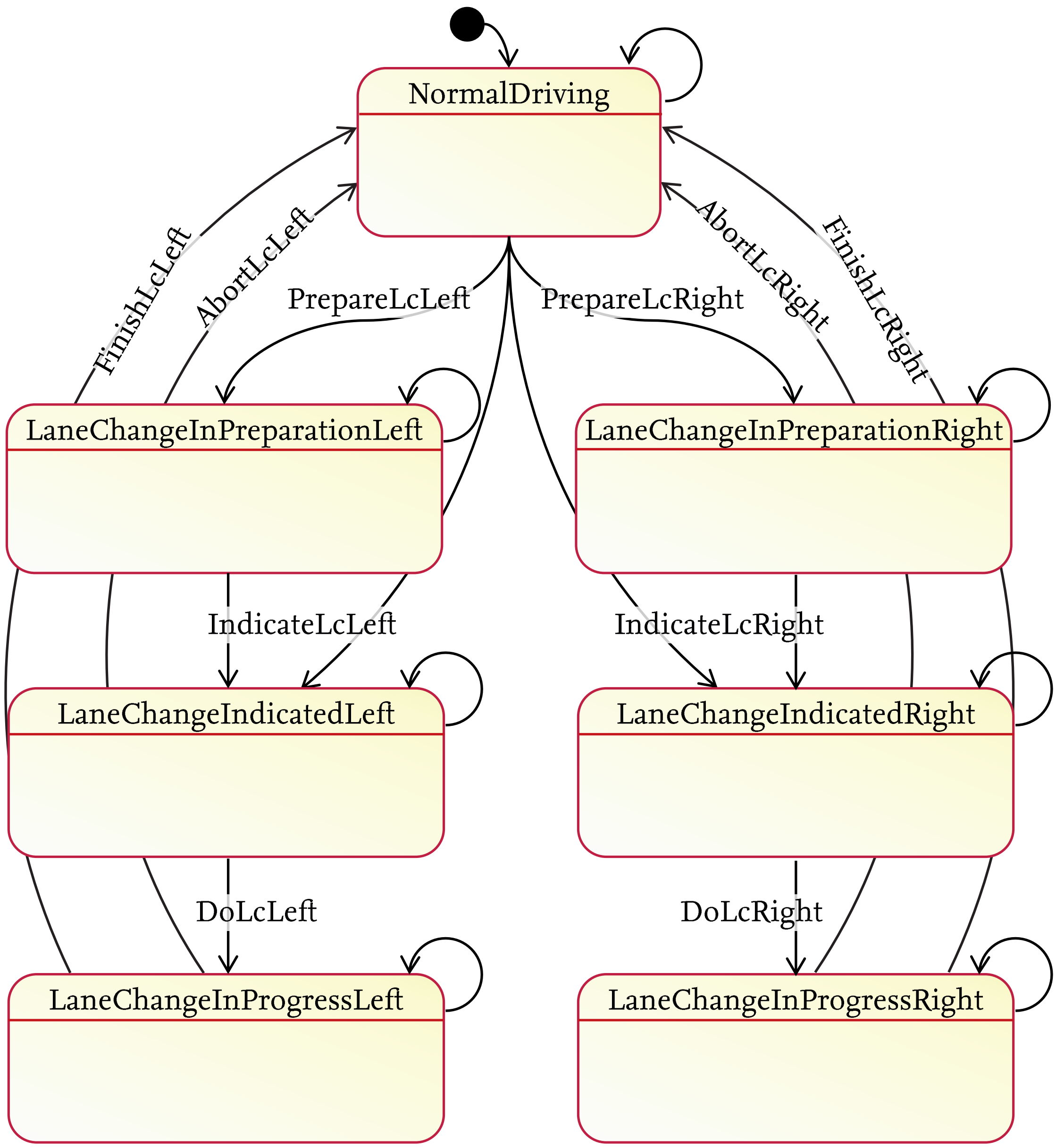


Figure 10.5: Simplified state transition diagram

Lane Change State

These lane change states are, first of all, *NormalDriving* for regular straight driving within a lane. Moreover, it includes lane changes to the left and right respectively: *LaneChangeInPreparation* if a gap adjustment (cf. section 10.3.3) is performed without flashing an indicator, *LaneChangeIndicated* if the indicator is flashed to announce a lane change or to prepare for a lane change with more exigent gap adjustments, and *LaneChangeInProgress* when a lane change is actually executed (building up lateral offset). Last of all, the lane change state may include *LaneChangeSuggested* for suggesting lane changes to a human driver and waiting for his clearance as in Ulbrich & Maurer (2013). For fully automated driving without a human driver in the loop, the *LaneChangeSuggested* state is unnecessary.

Actions

Depending on the system state, it is possible to select distinct actions. First and foremost, an action entails a value-discrete action type. Similar to the "lane change status", these include *DriveNormal*, and to the left and right, *PrepareLaneChange*, *IndicateLaneChange*, *DoLaneChange*, *AbortLaneChange*, and *SuggestLaneChange*. Moreover, an action entails an intended longitudinal delta position Δs_{pos} as an allowable sampling range for that delta position $s^-_{gap\ interval}$, $s^+_{gap\ interval}$, a longitudinal delta velocity Δs_{vel}, and a symbolic gap position index to be targeted. Depending on the successful completion or the abortion of a lane change, there are two transitions from *LaneChangeInProgress* to *NormalDriving* (cf. Figure 10.5).

10.2.2 Problem Specific Simplifications

Problem Setup

Section 5.2.3 presents partially observable Markov decision processes (POMDPs) as a general framework for decision making. Based on this, section 10.1 introduced the aspect of online and offline planning. Traditional POMDP-solution techniques require enumerable, value-discrete state spaces, observation spaces, and actions. If system states are not tractable to be enumerated, the other far less frequently used approach is to iterate over actions or policies (sequence of actions) in a belief tree as in Figure 10.4.

Problem -Specific Simplifications

Luckily, a lot of domain knowledge can be incorporated in the action selection process to tailor the decision process to the particular issue. There are several factors that render action iteration-based approaches far more versatile for tactical decision making in automated driving than it is in other domains:

- Planning horizons are relatively short: It is typically not possible to make long-term predictions of more than maybe 10 seconds. Hence, there is no need for a high, possibly infinite planning depth T.
- Action alternatives are relatively limited: Typically there might be several variations of the same maneuver but only very few mutually exclusive, discrete action alternatives. It might be possible to merge in front or behind a given vehicle on a neighbor lane, or even if traffic rules permit overtaking a vehicle in front on the left or right neighbor lane, or do not overtake a vehicle at all, but all in all the set of possible actions is rather small.
- Mixed observability: Some internal states, e.g., whether a lane change is in progress already or not, are completely known and thus free of uncertainty. Hence they reduce the model complexity by a lot and might even rule out some action alternatives.
- Limited planning accuracy is needed for actions in the far future: After all, only the immediate next action will be propagated to the subsequent modules, e.g., for trajectory planning. Hence, there is no need for a detailed plan on how fast to re-center in the neighbor lane given the vehicle is still at the stage of deciding how long to set the indicator to initiate a lane change maneuver.

- Situation prediction models can be non-linear: Situation prediction models tend to have discontinuities. E.g., a vehicle might be following a linear movement model most of the time, but it will not be able to tunnel through a slower vehicle in front and most likely it will also avoid moving into other vehicles. This renders partial derivative-based approaches from the model predictive control community intractable; action-based iteration can cope with that.

Drawbacks

Though all those advantages come at the cost of not being able to use most of the state of the art POMDP-solvers, and likewise, most of the ideas behind them (reachable state sets, etc.). Moreover, if a model with a finite state set is found, the state prediction model is a simple $X \times Z \times X$ matrix. Hence, it is easily possible to learn such a state prediction model. Learning such a model will be much harder for a more general $x_{t+\tau} = f(x_t, z_t, u_t, \tau)$ situation prediction model.

10.2.3 Tree-Based Policy Evaluation and Belief Tree Simplifications

Tree-based policy evaluation methods as in section 10.1 use the tree of beliefs b and actions u that can be obtained and executed in each time step and perform a forward search in these trees (Kochenderfer, 2015, p. 150 ff.). Figure 10.4 and 10.6 illustrate such belief trees. Based on a current belief b_0 at a time step t_0 several actions $u^{(0)}, u^{(1)}$ can be executed and result in a reward $r(b_0, u^{(0)}), r(b_0, u^{(1)})$ in Figure 10.6. Given a certain action was executed, new observations could be made and will result in new beliefs $b_1, ..., b_5$ at time step $t_{0+\tau}$. These new beliefs are once more the root nodes of some subtrees for this new time slice. The belief tree will be planned until it spans T time slices to cover the entire planning horizon.

Tree Complexity

As introduced in section 10.1, the tree size of a belief tree grows with the number of value-discrete actions $|U|$ and observations $|Z|$ to the power of the planning depth T: $O(|U|^T |Z|^T)$. The tree size of a DESPOT tree grows with $O(|U|^T K)$. For many application domains, this renders tree-based policy evaluation intractable, yet for lane change planning it is feasible as outlined in section 10.2.2.

Tree Simplifications

As an approximation, it is possible to consider only the most likely observation vector and the subsequent belief distributions as introduced in section 10.1. This tends to rule out situations where a future observation at a certain time step in the planning horizon will result in discrete outcomes that cannot be modeled by *one* parametric description of random distributions of the state variables to form a subsequent belief. An example for such a situation might be where another driver is predicted to let go of his individual comfort advantage and decide to decelerate in order to let an automated vehicle merge. By this simplification it is only possible to predict whether the other driver allows the automated vehicle to merge *or* not; not to evaluate *both* branches of letting merge with probability p and not letting merge with probability $1 - p$. In fact, such an approach has been used by Webster (2007, p. 39 ff.) for implementing lane change behavior in a simulation environment in a simplified domain without uncertainty. He uses a breadth-first search as in Russell & Norvig (2009, p. 81 ff.).

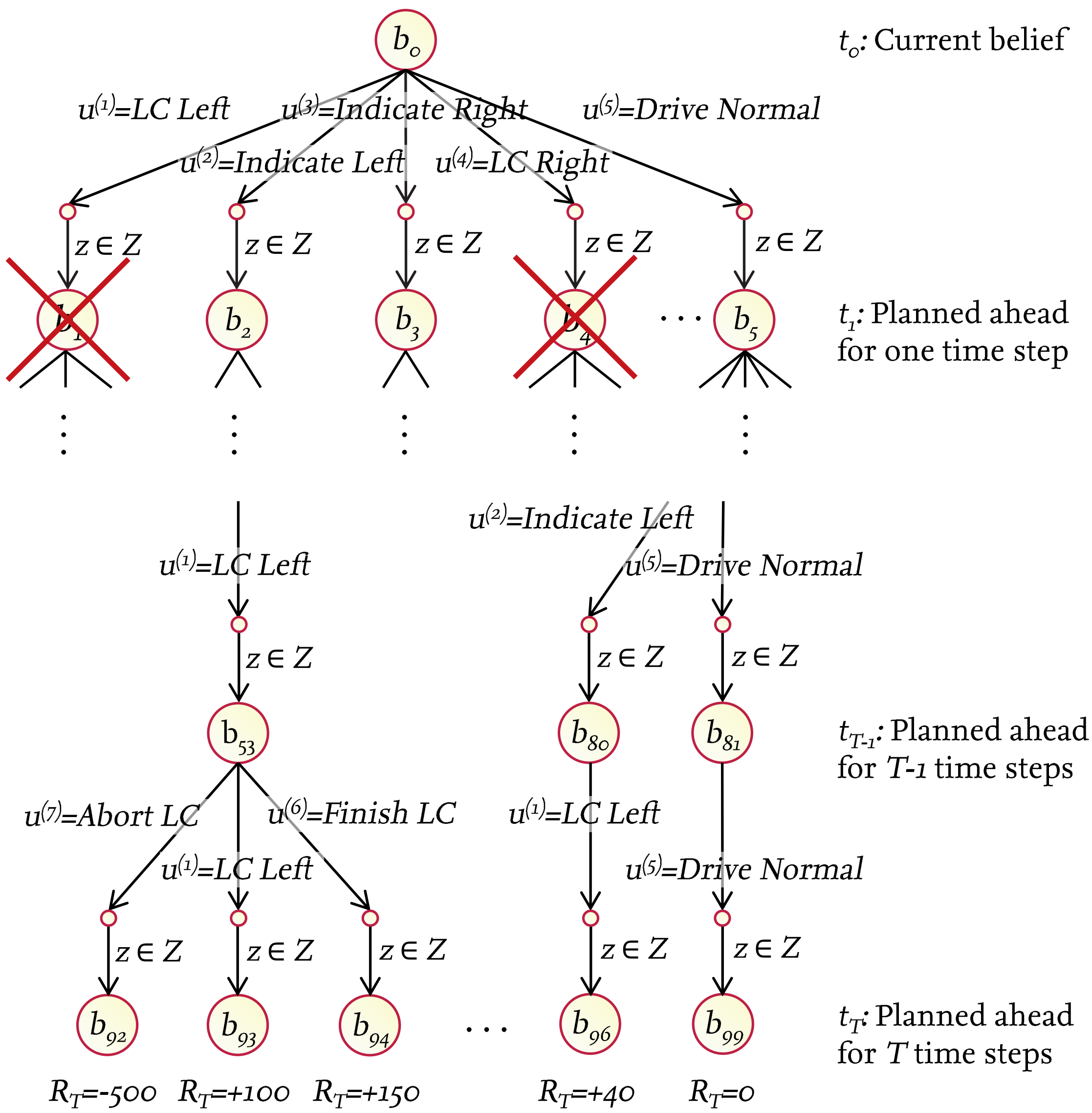


Figure 10.6: Belief or policy tree of beliefs and actions (LC = Lane change, R_T = Reward after T time steps, b = belief)

Figure 10.6 shows a second tree simplification for pursuing action $u^{(1)}$ while having belief b_0. This reflects the idea of constraints in model predictive control to rule out inefficient or infeasible solutions. Here no lane change should be executed without prior flashing of the indicator. In fact, it is possible to prune the tree even further by ruling out actions that would result in *unreasonable* policies (sequence of actions). E.g., if a lane change was decided it will be an unreasonable policy to abort a lane change and reinitiate a second lane change in two consecutive time steps. This helps to reduce the tree complexity. Every path to a fringe node at the end of the planning horizon at time step $t_0 + T$ will be a possible – to a certain degree – reasonable policy. Both tree simplifications reduce the number of policies to be evaluated by a lot as demonstrated in section 15.1.

Tree Pruning

10.2.4 Multi-Resolution Planning

Temporal Resolution

Another key idea to render online solution methods tractable for behavior planning is to incorporate the concept of multi-resolution planning. In fact, it is only necessary to plan with high temporal accuracy in the immediate future. This approach has been presented in some publications from the model predictive control research community.

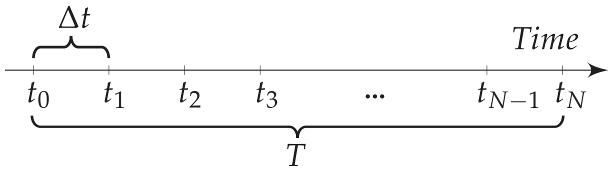


Figure 10.7: Fixed planning time steps

Time Discretization

Munos & Moore (2002) present concepts for non-uniform discretization of state spaces for solving problems, which are continuous in time and space. Earl & D'Andrea (2005) use an iterative refinement pattern for robot path planning with mixed integer linear programming. Culligan (2006, p. 26) extends this idea by using a variable time discretization for the optimization problem for a trajectory planning application for unmanned aerial vehicles. According to the nomenclature introduced earlier in section 10.2.3, a decision is based on information at the current time step t_0. In a planning ahead approach, the planning shall cover the planning horizon T, thus considering the future until $t_0 + T$. To obtain a computer solvable optimization problem, time is discretized into N time steps by $\Delta t_n = t_n - t_{n-1}$ for $n = 1..N$.

Fixed Temporal Resolution

A common way to discretize the planning horizon T is to use a *fixed* time step $\Delta t_n = \Delta t = T/N$ such that:

$$T = \sum_{n=1}^{N} \Delta t_n = \sum_{n=1}^{N} \Delta t \tag{10.1}$$

Variable Temporal Resolution

However, it is possible to cover the same planning horizon T with less planning time steps by using a variable resolution approach. In the far future, it is not necessary to plan with a high temporal resolution at all because a) predictions will be inaccurate anyway and hence detailed planning will be infeasible at any rate and b) although planning is done up to a certain time horizon, only the immediate next action of the planned policy will be executed and thus given to the subsequent modules. A certain degree of planning inaccuracy in the far future will only affect the current action by the extent to which this temporal resolution of action selection has an influence on the action that will be executed in the immediate next time step.

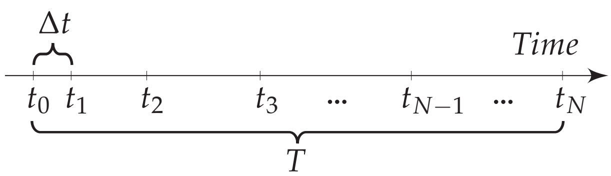


Figure 10.8: Variable planning time steps

For this thesis, a time step pattern of $n \cdot \Delta t = n \cdot 500\,\text{ms}$ is used. It results in the following sequence of time steps until the planning horizon $T = 14\,\text{s}$:

$\{0.5\,\text{s}, 1.0\,\text{s}, 1.5\,\text{s}, 2.0\,\text{s}, 2.5\,\text{s}, 3.0\,\text{s}, 3.5\,\text{s}\}$

10.3 Measurement Model [4]

Observable and Hidden State Variables

The task of the measurement model is to translate observations about the vehicle's ego state and the environment into an aggregated belief of the system. Observations may have value-discrete aspects, e.g., the number of lanes on the road the vehicle is currently driving on, or value-continuous ones, e.g., the distance to a front vehicle. The measurement model translates these into a belief of the system state, only containing the decision relevant state variables. Mathematically it is the conditional probability $p(z_t|x_t, u_t)$ for an observation z_t, given the system state x_t (Thrun et al., 2005, p. 149). The system state may be to some extent observable (e.g., it is known for how long the indicator might have been switched on in the immediate past), and partially hidden. Hidden state variables contain information about whether a lane change seems possible or beneficial in the current situation, which gap might be the best to head for to change to the neighbor lane, etc. Information for which no better state estimate can be achieved by modeling it as *hidden* will be modeled as *observable*. For instance, the distance to an object in front of the automated vehicle will never be completely known. However, earlier stages in the environment perception already came up with a best possible state and variance estimate. Hence, it is unlikely any improvements will be gained by assuming that the probability distribution for that distance is essentially inaccurate. The behavior planning algorithms will rather assume that the probability distribution of such a distance is the best possible information one could use for decision making.

Information Abstraction in Dynamic Bayesian Networks

Figure 10.11 shows the different stages within the measurement model. The left-most part of the image shows a visualization of the information in the context model as an abstract scene description of the vehicle itself and its environment. The next part of Figure 10.11 illustrates a situation abstraction of the lane change relevant information. To obtain beliefs for the distributions of hidden state variables, a dynamic Bayesian network is used.

The measurement model $O(x, u, z) = p(z|x, u)$ gives the probability of observing z if action u is performed and the resulting state is x. For this thesis, the goal is to build a belief tree as in section 10.1. Hence, it is necessary to come up with a state belief $b(x(t_n))$ and a predicted future observation for a situation prediction $b_{t+\tau} = f(b_t, z_t, u_t, \tau)$. Similar as in a hidden Markov model, a non-linear function $b(x(t_n)) = g(z(t_n), u(t_n), b(x(t_{n-1})))$ is used for both purposes. In some applications $g(z_t, u_t, b_{t-1})$ may be a low-dimensional linear equation. In this thesis, $g(z_t, u_t, b_{t-1})$ is unfortunately not a simple mathematical function but rather a procedural algorithm of 6171 lines of code. It is both, more generic and relevant *what* is calculated in $g(z_t, u_t, b_{t-1})$ rather than its detailed implementation. This *what* illustrated in Figure 10.10. The first step is the situation modeling from Chapter 8, followed by a high-level tracking explained in section 10.3.5 to improve consistency.

[4] Parts of this subchapter have been pre-published by the author in Ulbrich & Maurer (2015a).

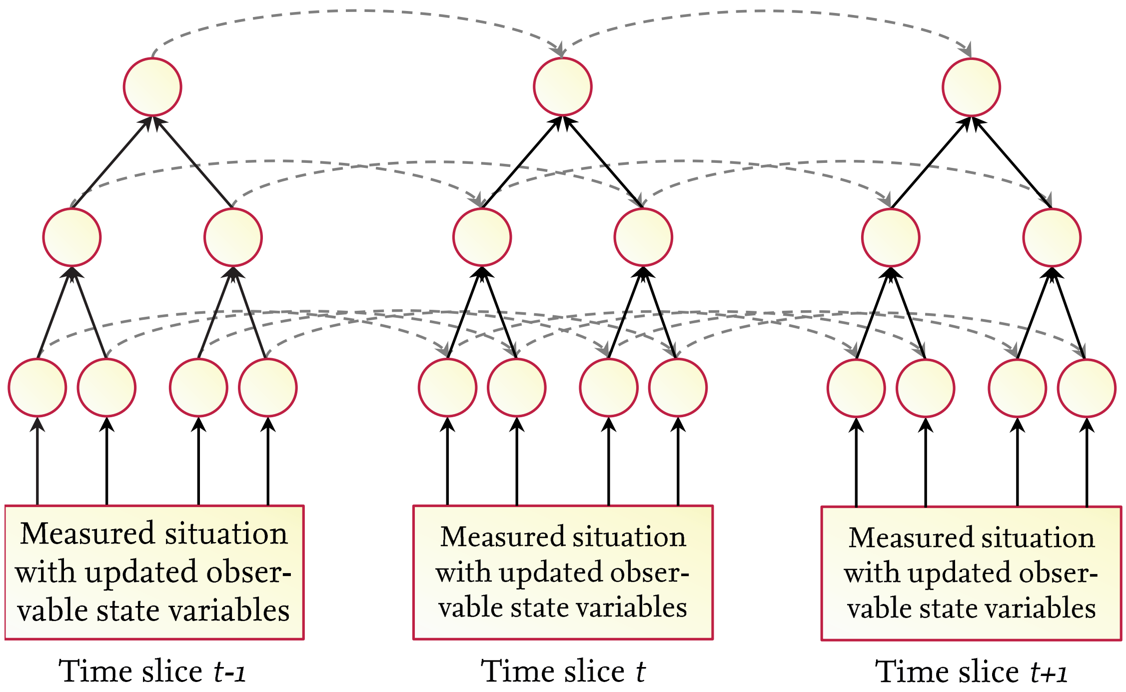


Figure 10.9: Measurement update for hidden state variables (circles) in a dynamic Bayesian network for lane change planning based on observable measurements (rectangular) and previous state estimates for a former time slice *t-1*. Arrows indicate information flow not necessarily conditional probability.

As a next step, a unscented transform with a subsequent probability distribution parameter estimation (cf. section 10.3.4) is performed using the dynamic Bayesian network.

Belief Update

Figure 10.9 illustrates a belief update in this dynamic Bayesian network.[5] Every round node is a hidden state variable in the dynamic Bayesian network. A new belief estimate at time slice t is derived from the latest values of observable state variables like the distances or velocities of objects and – with a certain weight – the old belief of that particular state variable at the previous time slice.

Tailoring Towards Lane Changes

The four highest-level hidden state variables for planning lane changes are whether a lane change is possible to the left/right and whether a lane change is beneficial to the left/right. To obtain state estimates for those hidden state random variables, several other random variables need to be estimated. The following sections will describe the necessary aspects to derive state estimates for those random variables. The measurement model is completed by a gap quality estimation and target point selection described in section 10.3.3.

10.3.1 Lane Change Beneficial Estimation

A lane change beneficial situation assessment relates to the deliberation phase in Heckhausen & Gollwitzer's (1987) rubicon model of decision phases (cf. section 5.1).

[5]Technically the conditional probability $p(z_t|x_t, u_t)$ provides a probability of an observation z_t, given a system state x_t and action u_t. Hence, the arrow in Figure 10.9 is often shown reversed. Here it faces upwards to represent the information flow.

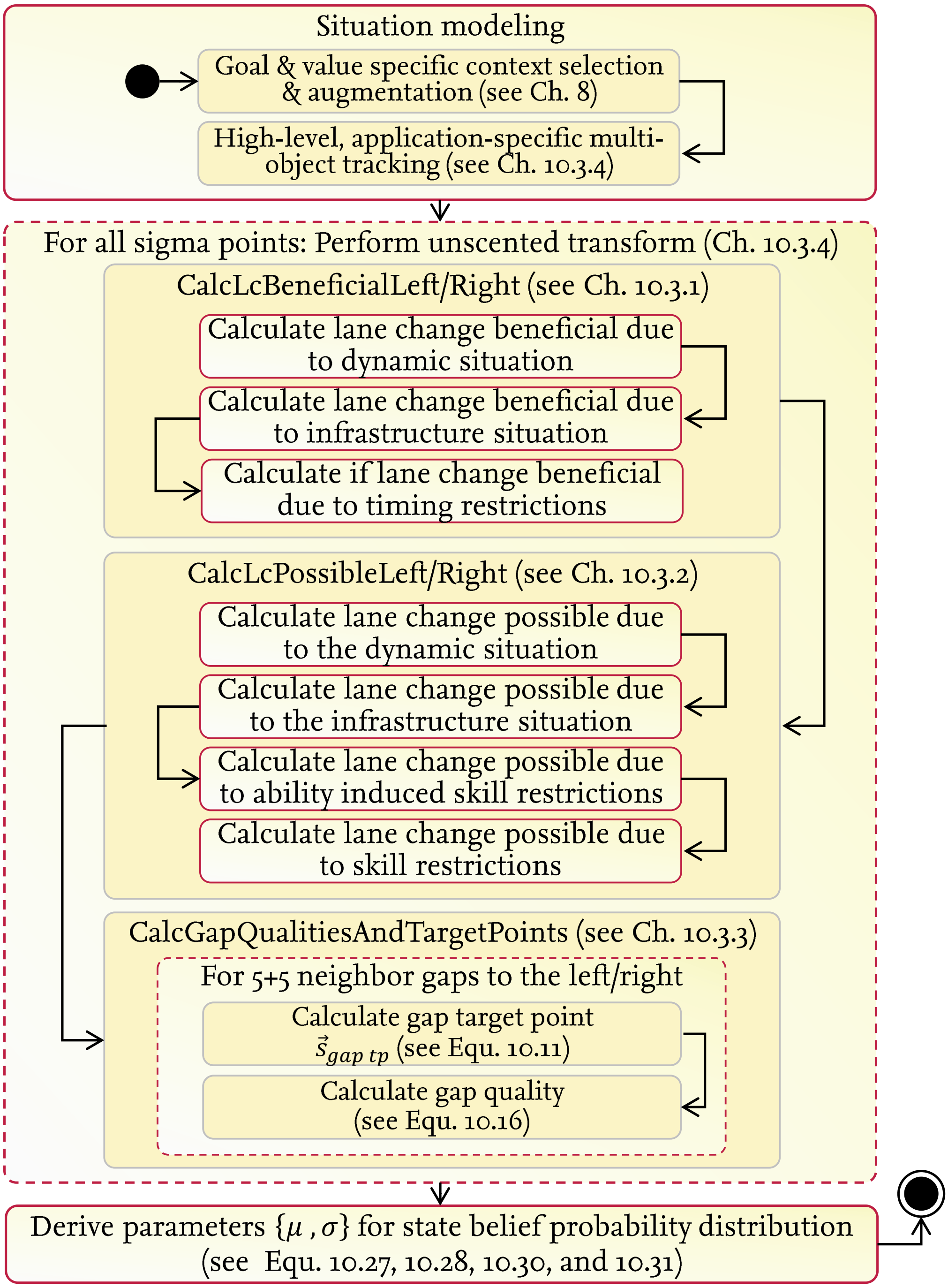


Figure 10.10: Simplified illustration of the measurement model and it's algorithm $g(z(t_n), u(t_n), b(x(t_{n-1})))$

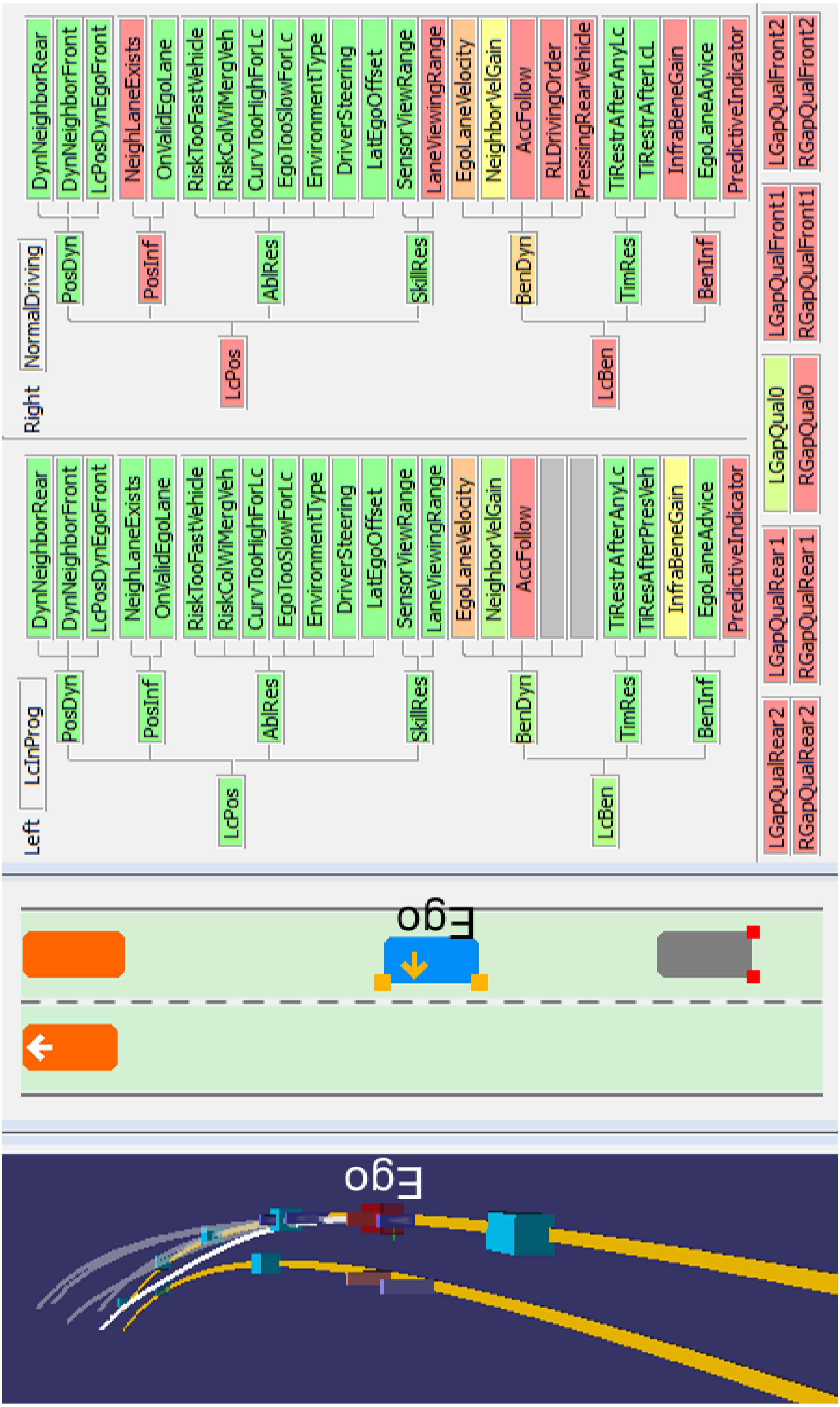


Figure 10.11: Situation for a lane change as it is perceived from the environment perception modules, abstract situation representation, and dynamic Bayesian network for lane change situation assessment. Variable abbreviations are explained in attachment C

It reflects the motives behind performing a lane change no matter if it is currently possible or not.

The lane change beneficial estimation is based on the idea of an "intended velocity" $s_{vel,intended}$ as in Wiedemann (1974, p. 10), Brilon & Brannolte (1977, p. 10 ff.), and Kopf (1993, p. 47). They define the intended velocity as a free-flow velocity without obstructing traffic $s_{vel,intended} = s_{vel,target} - \Delta s_{vel,infrastructure}$, as a "target velocity" minus a delta velocity to reflect the (infrastructure) situation specific velocity adaptations.[6] Intended Velocity

Rothengatter (1988) explains that the target velocity $s_{vel,target}$ for human drivers is based on motivational factors like "pleasure in driving", "traffic risks", "driving time", and "expenses". For this thesis, the target velocity for automated driving is simply derived from speed limits from a map/traffic signs capped by a maximum speed for the automated vehicle. For the scope of this thesis, the target velocity is not reduced to consider higher traffic risks while driving in rain or snow. For higher levels of automation this seems necessary as a part of a skill and ability monitoring to ensure safety in all conditions within the operational driving domain. Target Velocity

The current velocity may be lower than the intended velocity if the ego vehicle is obstructed by traffic. This is reflected by a delta velocity to consider slowdowns by dynamic elements around the ego vehicle $\Delta s_{vel,dynamic}$. Brilon & Brannolte (1977) states that an urge for overtaking arises when this delta velocity is higher than an "overtaking tolerance". According to them, this overtaking tolerance is influenced by the traffic density and individual driving style of a (human) driver. Obstruction by Traffic

The overall concept of lane change beneficial estimation presented in this thesis follows the concept of Schakel et al.'s (2012) "desires" to reflect incentives for performing lane changes. In this thesis, those "desires"[7] are structured into these "due to the dynamic situation", "due to the infrastructure situation", and "due to timing restrictions". The first is to consider the dynamic situation in different regions of interest (ROI) around the ego vehicle. The second to reflect if a lane change is beneficial due to infrastructure related information, e.g., if the navigation layer suggests making a right turn soon it could be beneficial to change lanes to the rightmost lane. A third aspect for lane change beneficial situation assessment is timing restrictions. For instance, if an automated vehicle has just made a lane change to the right to clear a lane for a pressing rear vehicle, it is not beneficial to go directly back in front of that vehicle for which the lane was just cleared. Hence, a timing restriction could help to reflect such behavior that a human driver would incorporate by situation specific reasoning.[8]

[6] Kopf (1993, p. 47) calls this delta velocity for (infrastructure) situation specific velocity adaptations simply v_S. For linguistic clarity, it was renamed to $\Delta s_{vel,infrastructure}$.

[7] This could also be called "motives" as in Puca (2016) and Heckhausen & Heckhausen (2008, p. 297 ff.).

[8] One may criticize that the timing restrictions are rather a surrogate than a profound technical solution. To the author this is right, yet a helpful path to avoid the complexity of true reasoning (cf. Chapter 11). To the author it is not a limitation of the snapshot concept of a situation, because of the very reason that timing restrictions *can* be modeled as part of the situation aspects.

Lane Change Beneficial due to Dynamic Situation

Unobstructed Overtaking and Follow Velocity

Kopf (1993, p. 86) presents an approach to estimate a (human) driver's target velocity and a so called "unobstructed overtaking velocity" and "unobstructed follow velocity" as velocities for when a (human) driver prefers to follow a front vehicle (slightly below target velocity) and which he would pick to overtake (slightly above target velocity). This idea is not followed upon, because in automated driving the target velocity is known by definition and the overtaking velocity shall by definition not exceed the target velocity (no speeding in automated driving).

Speed Advantage

For simulating driving behavior, Gipps (1986) presents a decision tree for whether a lane change is beneficial due to the dynamic situation or not. This entails aspects like whether a heavy vehicle is in a certain lane or if there is a slow preceding vehicle. To evaluate the impact of preceding vehicles, the "speed advantage" for each lane is calculated. A lane change is considered beneficial if a velocity gain of 1 m/s is achieved.[9]

Politeness

Kesting et al. (2007) and Schakel et al. (2012) use the intelligent driver model (cf. section 6.2) to calculate a possible acceleration for the ego vehicle in case of performing a lane change and not performing a lane change. If a lane change enables an acceleration, it contributes to the "desire to gain speed" (Schakel et al., 2012). Likewise, the positive or negative acceleration difference is calculated for the current follower-vehicle and a possibly new follower vehicle after a lane change of the ego vehicle. The impact for improving the acceleration for the old follower vehicle and possibly negatively affecting a new follower vehicle is weighted with a "politeness factor" and added to an ego vehicle acceleration gain. If it is higher than a threshold, a lane change is considered beneficial due to the dynamic situation. In this thesis, the politeness to clear a lane is reflected in the consideration of pressing rear vehicles as part of the predecisional deliberation as in section 5.1. The discomfort for other vehicles in a lane change execution will be considered as part of the lane change possible situation assessment in the planning phase of the Rubicon model (cf. sections 5.1 and 10.3.2).

Keep Right

Another aspect considered by Gipps (1986), Kesting et al. (2007), and Schakel et al. (2012) are asymmetric traffic rules. In many European countries it is necessary to follow a right lane driving order. Kesting et al. (2007, p. 89) incorporates an "acceleration bias" to reflect a right lane driving order. Likewise, Schakel et al. (2012) incorporates a "desire to keep right". In their model, it is set to a fixed numeric value as a bias for driving right. They inhibit the keep right incentive if there is slow traffic on the right lane or if a lane change right contradicts a desire to follow the route.

Queuing Advantage

Hidas (2002) uses the Gibbs-model as a foundation for lane change beneficial evaluation of the dynamic situation. He adds the aspect of a "queuing advantage" for evaluating if a lane change is beneficial. This reflects the advantage of being further in front if there are lane specific queues of different length, which is often found

[9] If asymmetric lane change rules are necessary for a particular country, Gipps suggests to consider a lane change beneficial even at a velocity gain of -0.1 m/s to implement something like a right lane driving order.

at traffic lights. In this thesis, no distinct considerations for queuing advantages will be made. If traffic moves with different velocities on different lanes in a traffic jam it will result in an appropriate benefit gain. Since highways are the main focus in this thesis, the aspect of queuing is dispensable. Yet, it may be a worthwhile addition for urban driving.

In this thesis, any kind of beneficial considerations for performing a lane change regarding the dynamic situation are based on estimated lane specific traffic flow velocities and resulting velocity gains. A lane specific traffic flow velocity (short: lane velocity) is calculated from the low pass filtered object velocities of vehicles driving on that particular lane.

No Speeding

The neighbor lane velocities will be capped by a $min(LaneVelocity, SpeedLimit)$-operation, because the automated vehicle should not see a benefit[10] in overtaking a vehicle that is already driving faster than a road's speed limit.[11] The speed limit entails an infrastructure related speed limit, or one given by the abilities or target velocity of the automated vehicle.

If the ego vehicle is in adaptive cruise control following mode, due to a slower front vehicle, it should try to perform a lane change to obtain a velocity gain. A possible slowdown due to a vehicle is weighted by that vehicle's existence uncertainty (cf. section 2.5). On a German highway, where overtaking is only allowed on the left, this lane change should be on the left. Hence, the left neighbor lane velocity will be set to equal the speed limit.[12]

Right Lane Driving Order

If there is no reason to remain left, e.g., in the domain of a German highway, the automated vehicle should move to the rightmost lane to follow the right lane driving order. Vice versa, if there is a slower vehicle on the right neighbor lane, the automated vehicle should only merge behind it if it is sufficiently far away. Therefore, the time gap and time-to-collision towards such an object are calculated and evaluated. For the time-to-collision, different thresholds are used depending on whether there is a vehicle behind the ego vehicle ($TTC_{threshold} = 25\,\text{s}$), or whether there is no rear vehicle ($TTC_{threshold} = 35\,\text{s}$). [13]

Pressing Rear Vehicle

If the automated vehicle is driving on the leftmost lane on a highway with a right lane driving order and a rear vehicle is driving with a very small time gap behind the automated vehicle, it is assumed to have the intention of overtaking the automated

[10] This is to prevent the automated vehicle from attempting to overtake but then being unable to overtake due to following the speed limit.

[11] Details of the implementation are presented in the disclosure of the invention "Ermitteln von Fahrstreifenflussgeschwindigkeiten zur Bewertung des Nutzens von Fahrstreifenwechseln in dünnem Verkehr (English title: Determining lane specific traffic flow velocities to evaluate the benefit of performing lane changes in sparse traffic)" (Ulbrich et al., 2015a).

[12] This aspect ensures that the automated vehicle rather sticks to the left lane in similarly slow-moving traffic on several lanes. This is done in the anticipation that the left lane is more likely to move faster once the traffic clears up. Yet, the automated vehicle is also allowed and will overtake on the right at velocities below 60 km/h.

[13] Details of the implementation are presented in the disclosure of the invention "Verfahren zur Befolgung des Rechtsfahrgebots bei der Bewertung des Nutzens von Fahrstreifenwechseln (English title: Procedure for adherence of the right lane driving order rule in the benefit evaluation of lane changes)" (Ulbrich et al., 2015e).

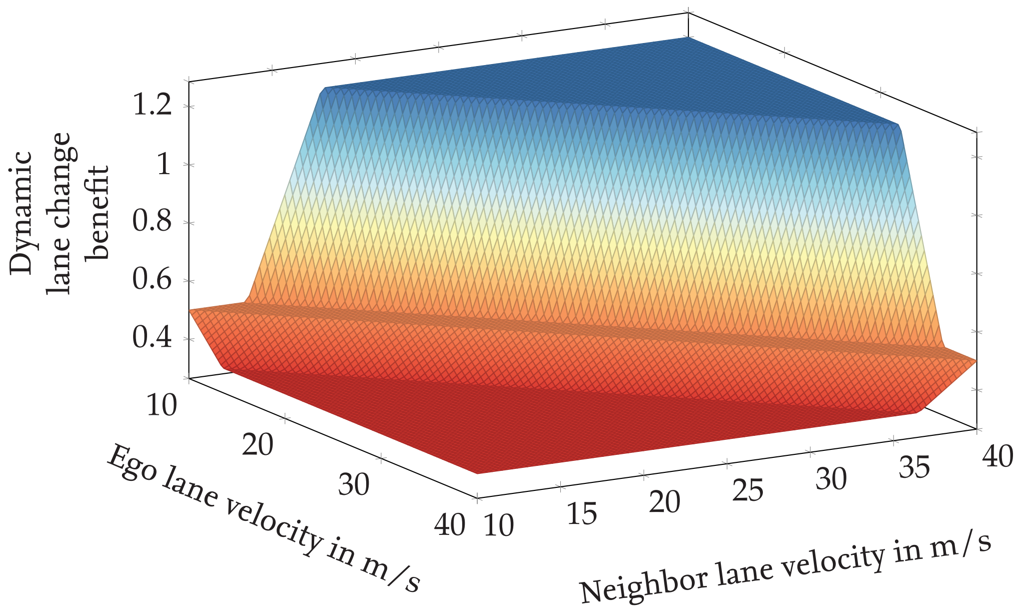


Figure 10.12: Lane traffic flow velocity gain transfer function

vehicle. Hence, a human driver would be willing to move over to the right lane even if he might be slowed down a bit by a slower vehicle on that lane. To imitate that behavior in an automated vehicle, the aforementioned $TTC_{threshold}$ for accepting a right neighbor lane will be lowered to $TTC_{threshold} = 10\,\mathrm{s}$. [14]

Velocity Gain

After all those intended behavior based lane velocity modifications, a velocity gain is calculated by a transfer function as in Figure 10.12. It uses the difference between the neighbor lane velocity and the ego lane velocity and caps it with a maximum and minimum dynamic benefit gain. If only a small positive velocity gain is obtained from a lane change it will be hidden by the transfer function in Figure 10.12. Likewise, a very high or low velocity gain is capped. Between those ranges linear transfer functions are used to calculate a dynamic lane change benefit. Attachment F.1 provides details on the implementation of the function in Figure 10.12.

Lane Change Beneficial due to Infrastructure Situation [15]

Lane Advice

Apart from dynamic traffic situation based reasons for a lane change, the predominant source for lane change decisions will be caused by infrastructure related factors. For instance, if the navigation layer necessitates a right turn to another road, the tactical layer should take care to reach the turning point on the rightmost lane.

Gipps (1986), Toledo et al. (2003), and Schakel et al. (2012) cover different infrastructure aspects in their lane change simulation models. If an intended turn is closer

[14] Details of the implementation are presented in the disclosure of the invention “Fahrstreifenwechsel bei Hinterfahrzeug mit Überholabsicht (English title: Lane changes with the consideration of rear vehicles with an intention of overtaking)” (Ulbrich et al., 2015d).

[15] The algorithms in this section have been developed in close cooperation with my colleague Christian Appelt who is responsible for the a-priori lane advice calculation resulting from the strategic layer routing algorithms.

than a certain distance, Gipps (1986) uses it to motivate a lane change towards a lane in that direction. Likewise if a lane is obstructed it will be considered less favorable depending on the distance towards this obstruction. Gipps (1986) uses rule-based distance thresholds where he ramps in the relevance of aspects. He distinguishes between three distances: 1) If an intended turn is close, the turning lane or an adjacent lane is selected. Gaining or maintaining speed is considered irrelevant. 2) If an intended turn is sufficiently far away, it has no impact on lane change planning. 3) If it is in between, gaining speed advantages from overtaking are ignored. Another point that is considered in his model to motivate lane changes because of infrastructure are special lanes like high-occupancy vehicle (HOV) lanes or bus lanes. The automated vehicle is allowed to enter those lanes temporarily, but is not allowed to remain there as soon as there is no longer a reason not to change to the target lane.

Gipps

Toledo et al. (2003) develop a probabilistic model to simulate discrete behavior choices. To calculate a lane choice probability, they use "lane utility functions" and a "driver-specific random term [...] capturing correlations between observations of the same driver over time". Infrastructure related parts of the lane utility functions are the distance to a next exit, and step functions to incorporate how many lane changes are necessary to get there and to reflect an averseness against driving on a rightmost lane to reduce exposure towards merging vehicles.

Toledo

Schakel et al. (2012) models the "desire to follow the route" as an infrastructure related desire. To follow a route, an incentive is derived if the current lane does not match the route. Depending on the ego velocity, the remaining time or remaining distance per required lane change determines that desire.

Desire to Follow Route

In this thesis, a similar approach is taken as in the literature. To reflect the desire to follow the route, every lane segment is attributed with a cost to reach the destination. These costs will be called *lane advices* in the reminder of this section. On a highway, all lanes will have a similar lane advice. However, if an automated vehicle proceeds closer to an exit it is supposed to take, the rightmost lane will get a higher lane advice than the other lanes. At the last possible point to leave the highway at that exit, the lane advice for the ramp will be very high, and the lane advice for driving straight on the highway will be close to zero on all lanes because it would cause a detour. Likewise, longterm lane blockages or lane usage restrictions are incorporated into the lane advice.

In this Thesis

To obtain a numeric value between zero and one for the infrastructure related lane change benefit, the transfer function in Figure 10.14 is used. The lane advices are calculated in such a way that the most beneficial lane to drive on has, by definition, a lane advice of 1.0. A less beneficial lane has a lane advice in the range of 0.0 to 1.0. Thus, typically the ego lane or a neighbor lane has a lane advice of 1.0. Thus, most of the time the lane advice transfer function goes along the red line in Figure 10.14. Only in rare cases, if the optimal lane is two or more lanes to the left or right, the red line is left. If the lane advice is close to 0.5 (neutral), small lane advice differences are "hidden" from the lane change decision making. This is useful to avoid a slight inclination towards driving on the rightmost lane only because the automated vehicle has to leave a highway for instance 10 km further down the road

Lane Advice Transfer Function

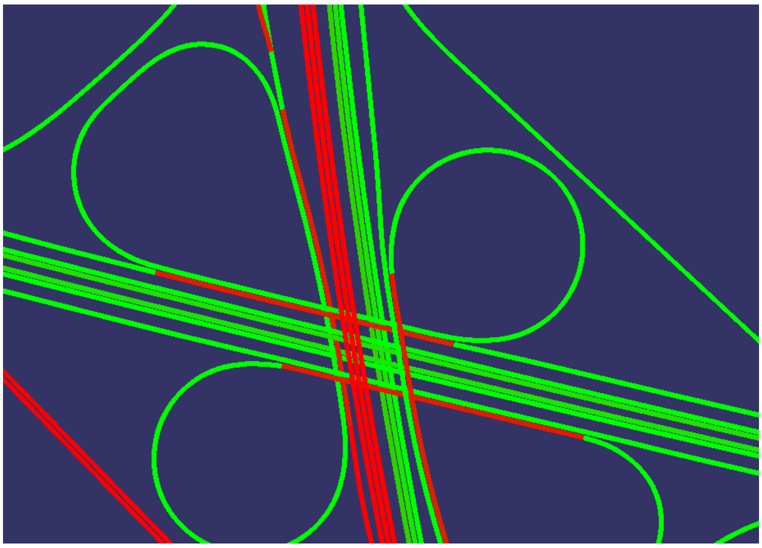


Figure 10.13: Lane advice from the navigation layer. Image courtesy of Christian Appelt

to a right exit ramp (cf. lane changes in free traffic). If the neighbor lane is more beneficial by 0.01 than the ego lane (optimal lane has jumped to neighbor lane), then the neighbor lane is assumed to be the optimal lane (lane advice = 1.0) even if it is just in the direction *towards* the optimal lane. This helps to make lane changing more aggressive if the optimal lane is more than one lane away.

Numeric Calculation

The infrastructure related benefit is then calculated by the proportion of the neighbor lane advice divided by the sum of an optimal lane advice (1.0) and the lane advice of the less than optimal ego or neighbor lane. If the denominator converges to zero and might cause numeric instabilities, an infrastructure related lane change will be discouraged. Attachment F.2 shows the full algorithm to obtain the transfer function in Figure 10.14.

Not Considered Aspects

The handling of HOV lanes as in Gipps (1986) have not been incorporated in this dissertation because they where of no relevance where the car was to be demonstrated. Likewise, no driver-specific lane change behavior has been incorporated. For the ego vehicle in automated driving, people inside the vehicle are considered as passengers and thus no learning of their preferences has currently been implemented. For other vehicles no identification of driver-specific lane change behavior is performed because the chance of seeing stochastically relevant driver-specific lane change behavior before a vehicle track falls apart is currently often too small given occlusions and the limited field of view from current environment sensors. Permanently avoiding a rightmost lane as in Toledo et al. (2003) has not been implemented as it contradicts the right lane driving order. Yet, the issue of cooperative behavior at highway on-ramps is specifically addressed and implemented as in sections 4.3 and 10.5.1.

Lane Change Beneficial due to Timing Restrictions

A third aspect to be considered for a lane change beneficial evaluation is timing restrictions. Certain timings render a lane change less likely to be beneficial, even if it is possible to execute it.

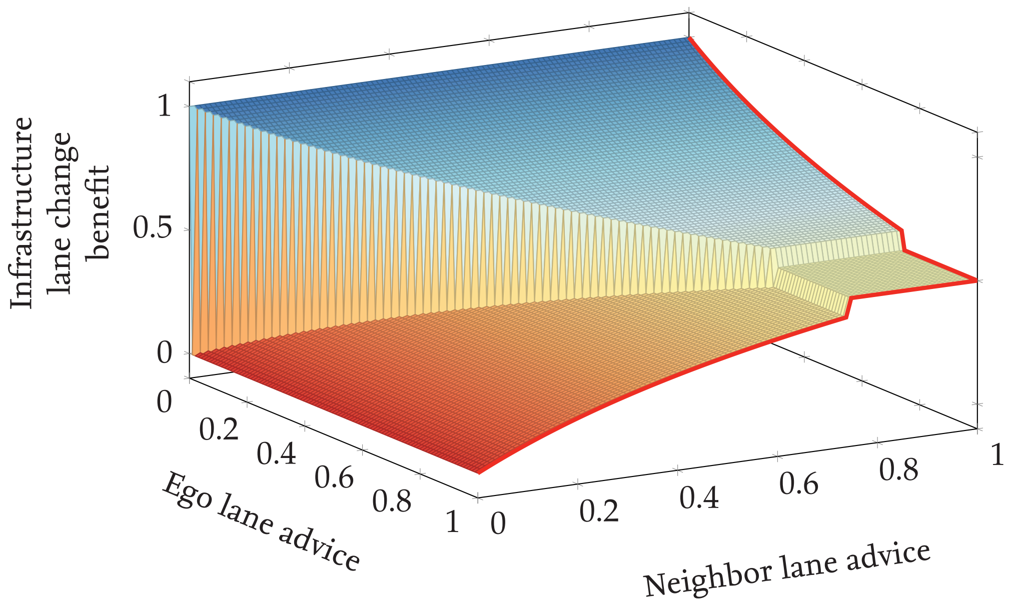


Figure 10.14: Infrastructure related lane change benefit as a function of the ego lane advice and the neighbor lane advice. The red line indicates the typically used part of the curve, where the optimal lane is either the ego or direct neighbor lane

First of all, a lane change is not considered beneficial if the time since the last successful completion of a lane change is less than $PreventLcAfterAnyLc = 0.5\,\mathrm{s}$. Further, this gives the lane tracking algorithms time to stabilize a tracking of lane markings and reduces false situation assessments due to not perfectly compensated pitch and roll movements of the ego vehicle during a lane change. Last of all, it makes lane changes less unsettling for passengers in the car as it gives them time to have a look at the traffic in the next lane if they intend to monitor the system. Likewise, it makes a lane change easier to supervise for a safety driver.

Recentering after Lane Change

Secondly, in domains with a right lane driving order (e.g., German highway), a lane change to the right is inhibited for $PreventLcRightAfterLcLeft = 5.0\,\mathrm{s}$ after a lane change has successfully been completed to the left. This is useful to avoid an automated vehicle initiating an overtaking maneuver by a lane change to the left and then directly changing back to the right lane because the vehicle to be overtaken temporarily drove faster or was at least perceived to drive faster. For countries without a right lane driving order on highways (e.g., the USA), this condition needs to be modified such that any lane change caused due to a dynamic lane velocity benefit is not revoked by a lane change in the opposite direction within, e.g., 5.0 seconds. The latter rule is more prone to errors as the dynamic velocity gain is an estimated quantity with high uncertainty. Moreover, using the latter rule on highways with high velocity differences between vehicles and lanes would result in getting stuck more easily behind slow trucks on a rightmost lane because lane changes back to the left lane would also be inhibited for, e.g., the 5.0 seconds.

Changing Back During Overtaking

Thirdly, if a lane is cleared for a pressing rear vehicle to overtake (cf. Chapter 4 and section 10.5.1), it is not beneficial or for a human driver logical, if an automated vehicle for whatever reason directly changes back in front of that pressing rear vehicle.

Pressing Rear Vehicle

This is potentially even hazardous because the driver of the pressing rear vehicle will not expect the automated vehicle to move back and thus he will just accelerate. Therefore, a lane change to the left on a highway is prevented if a lane change to the right was executed less than $PreventLcLeftAfterRearPressingVehicleLcR = 3.0$ seconds ago and was caused by a pressing rear vehicle. One should note that a pressing rear vehicle is only considered in domains with a right lane driving order.

Last Activation

Last of all, a lane change is not considered beneficial if it is planned less than $TimeSinceLastActivation = 5.0$ seconds after a handover scenario from a human driver towards the automated driving mode. This helps to give a human driver time to get used to the automated driving mode before a more complex maneuver like a lane change is initiated. This rule is only surpassed if a lane change needs to be executed in the next $TimeThresholdToAllowUrgentLc = 10\,\mathrm{s}$ in order to achieve the mission goal. This is particularly relevant to avoid blocking lane changes in highway interchanges or at intersections.

10.3.2 Lane Change Possible Estimation

A situation assessment of whether a lane change is possible relates to the planning phase in Heckhausen & Gollwitzer's (1987) rubicon model of decision phases (cf. section 5.1). To decide whether a lane change is possible it is necessary to estimate if a lane change is possible due to the dynamic environment in different regions of interest (ROI) around the ego vehicle and if a lane change is possible due to infrastructure related information, e.g., if there is a neighbor lane at all, if traffic rules/lane markings allow a change into that lane, etc. However, as illustrated in section 9.4 lane changes may induce additional risk towards automated driving than just driving within a lane. Hence, a lane change should only be possible if there are no skill or ability restrictions to prevent a lane change.

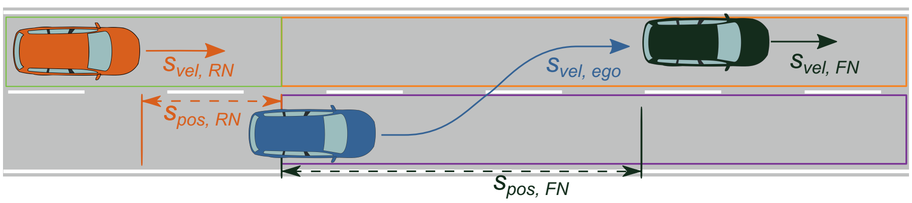


Figure 10.15: Lane change situation while a vehicle is approaching the automated vehicle (blue) from behind. Coordinate system defined in attachment D

Lane Change Possible due to the Dynamic Situation

ROIs for Dynamic Elements

Maybe the most obvious aspect for decision making about whether a lane change is possible is the consideration of dynamic elements in the automated vehicle's environment. To achieve this, the automated vehicle's environment is split into different regions of interest. Figure 10.15 illustrates three different regions of interest for deciding about a lane change to the left. In violet is the region of interest "front ego" (FE), in orange is "front left" (FL) and in green is "rear left" (RL). Accordingly, "front ego" (FE), "front right" (FR) and "rear right" (RR) will be considered for a lane change to the right. For simplification, from now on the left or right lane is considered as a neighbor lane with the regions of interest "front neighbor" (FN) and "rear neighbor" (RN), respectively. This nomenclature avoids redundancy for right-left mirrored evaluation tasks.

To calculate whether other vehicles allow a lane change, the motion equations of a point of mass similar to in Sparmann (1978, p. 19), Kopf (1993, p. 63), and Chen (2009, p. 24) are used.

$$s(t) = 0.5 \cdot s_{acc} \cdot t^2 + s_{vel} \cdot t + s_0 \tag{10.2}$$

$$s_{pos,ego}(t) = 0.5 \cdot s_{acc,ego} \cdot t^2 + s_{vel,ego} \cdot t + s_{pos_0,ego} \tag{10.3}$$

$$s_{pos,RN}(t) = 0.5 \cdot s_{acc,RN} \cdot t^2 + s_{vel,RN} \cdot t + s_{pos_0,RN} \tag{10.4}$$

$$\begin{aligned} s_{pos,ego}(t) - s_{pos,RN}(t) &= 0.5 \cdot (s_{acc,ego} - s_{acc,RN}) \cdot t^2 \\ &\quad + (s_{vel,ego} - s_{vel,RN}) \cdot t \\ &\quad + (s_{pos_0,ego} - s_{pos_0,RN}) \end{aligned} \tag{10.5}$$

By calculating the derivative of equation 10.5 and setting it equal to zero it is possible to determine when the distance to the rear vehicle is minimal. It is advantageous to enforce a minimal time gap T_{RN} towards a vehicle approaching from behind with a velocity difference of $\Delta s_{vel} = s_{vel,RN} - s_{vel,ego}$. Moreover, a reaction time to initiate a braking maneuver of T_R will be assumed for the driver in the orange vehicle approaching the blue automated vehicle in Figure 10.15. For the sake of analytic solvability of the motion equations, it is assumed that the driver of the orange vehicle will brake with a constant acceleration $s_{acc,RN}$. The distance that the orange vehicle needs to brake behind the blue automated vehicle can be calculated by:

$$s_{pos,RN} = \frac{\Delta s_{vel}{}^2}{2 \cdot s_{acc,RN}} - \Delta s_{vel} \cdot T_R - s_{vel,ego} \cdot T_{RN} \tag{10.6}$$

Solving for $s_{acc,RN}$, the negative acceleration enforced on the rear neighbor vehicle, yields:

$$s_{acc,RN} = \frac{\Delta s_{vel}{}^2}{-2 \cdot (-s_{pos,RN} - \Delta s_{vel} \cdot T_R - s_{vel,ego} \cdot T_{RN})} \tag{10.7}$$

This extends Sparmann's (1978, p. 19) "necessary deceleration" used in Kopf's (1993, p. 62 ff.) "obstruction model" by incorporating a reaction time T_R. For regular decision making, $T_R = 1\,\text{s}$, $T_{RN} = 0.8\,\text{s}$, $s_{acc,RN,begin} = -1\,\text{m}/\text{s}^2$, and $s_{acc,RN,abort} = -3.5\,\text{m}/\text{s}^2$ are used. $s_{acc,RN,begin}$ is the braking acceleration that a cooperative driver is assumed to allow another vehicle to merge in front of him. Hence, the automated vehicle starts initiating a lane change as long as no other driver has to brake more than $-1\,\text{m}/\text{s}^2$. The automated vehicle will abort a lane change if another vehicle has to brake more than $s_{acc,RN,abort}$ in order to prevent a collision.

Considering Rear Neighbor Vehicles

Figure 10.16 illustrates the braking accelerations necessary for other traffic participants in order to prevent a collision with a lane changing automated vehicle or to violate their safety time gap by such. It depicts the necessary braking acceleration

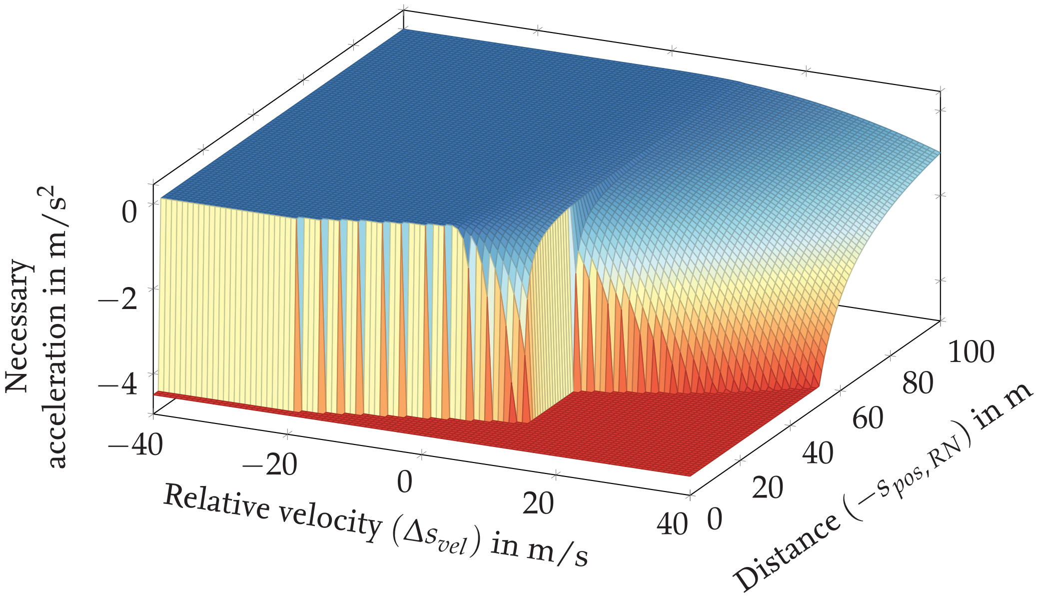


Figure 10.16: Necessary/accepted braking accelerations for a vehicle approaching an automated vehicle from behind

as a function of the relative velocity Δs_{vel} and the negative distance $s_{pos,RN}$. Just using equation 10.6 would not result in a hyperplane without the dent around low relative velocities. However, to ensure that lane changes will be performed in dense, constantly moving traffic with low relative velocities, it is necessary to accept much smaller time gaps than $T_{RN} = 0.8\,\mathrm{s}$. The following logic is used to calculate an acceptable time gap and causes the dent in Figure 10.16: [16]

$$T_{RN} = \begin{cases} 0.3\,\mathrm{s} & \text{if } \Delta s_{vel} < 3\,\mathrm{m/s} \\ 0.8\,\mathrm{s} & \text{if } \Delta s_{vel} \geq 3\,\mathrm{m/s} \end{cases} \tag{10.8}$$

$$T_{FN} = \begin{cases} 0.3\,\mathrm{s} & \text{if } \Delta s_{vel} < -3\,\mathrm{m/s} \\ 0.8\,\mathrm{s} & \text{if } \Delta s_{vel} \geq -3\,\mathrm{m/s} \end{cases} \tag{10.9}$$

Considering Front Neighbor Vehicles

Similar calculations can be done for approaching a slower vehicle in front. Figure 10.17 illustrates the resulting necessary braking accelerations as a function of the relative velocity difference and the distance. Again, the logic in equation 10.9 is used to calculate necessary braking accelerations. Once more, a lane change will be initiated as long as the necessary braking acceleration is below $s_{acc,ego,begin} = -1\,\mathrm{m/s^2}$ and a lane change will be aborted if it becomes less than $s_{acc,ego,abort} = -3.5\,\mathrm{m/s^2}$.

Apart from vehicles in the neighbor lane, it is necessary to consider vehicles in front of the automated vehicle in the ego lane and behind the automated vehicle

[16] These parameters have empirically been found suitable to reflect human-like driving behavior.

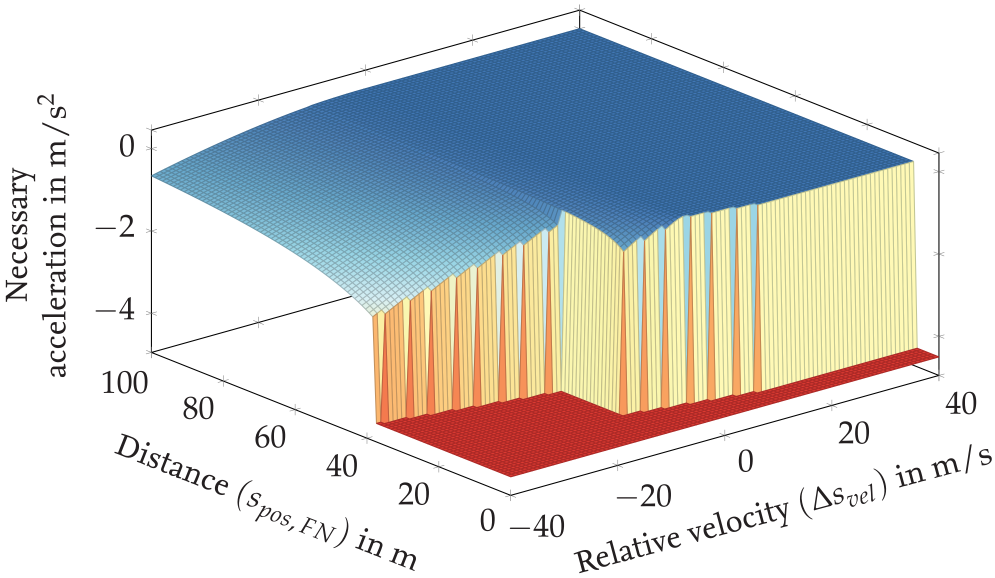


Figure 10.17: Necessary/accepted braking accelerations for an automated vehicle approaching a slower vehicle in front

in the ego lane. While the longitudinal control and trajectory planning should ensure distance keeping to the front vehicle, it may inhibit a lane change in special situations if no sufficiently large gap to the front vehicle can be ensured. This might happen in traffic jam situations as well as when the front vehicle has just merged into the ego lane and jerk-minimizing trajectory planning has yet to slow down to get to its intended time gap towards that front vehicle.

Considering Front Vehicles in the Ego Lane

Fast vehicles behind the automated vehicle in the ego lane are indirectly considered: Based on the assumption that they may change to another lane to overtake the ego vehicle while maintaining their speed, they are mirrored onto neighbor lanes and are thus also considered in the lane change possible evaluation. This mirroring is performed if their relative velocity is higher than *FastRearObjectVelocityThreshold* $=$ 10 m/s. This helps to accommodate situations where a faster vehicle from behind approaches the ego vehicle and changes lanes to overtake at the last moment. Without such consideration of these vehicles, the automated vehicle might initiate a lane change by itself and would block those fast rear vehicles, or would initiate a lane change and later abort it during its execution because the fast rear vehicle has changed to the neighbor lane during the course of the lane change.

Considering Fast Rear Vehicles in the Ego Lane

All measurements from the environment perception modules come with some inherent uncertainties (cf. section 2.5). For all distances, velocities, and accelerations Gaussian distributions are assumed with $N(\mu, \sigma)$ as a random process illustrated by Figure 10.18.[17] To be on the safe side[18] for lane change behavior planning, every random variable is conservatively estimated, by using not just the mean but rather

Measurement Uncertainties

[17]For existence uncertainty a Bernoulli distribution is assumed.

[18]This can only account for the *measurement uncertainty* as part of the *perception uncertainty*. Such a Gaussian distribution cannot consider the *existence uncertainty* or the *association uncertainty* introduced in section 2.5. They are indirectly considered by their effect on the *measurement uncertainty*.

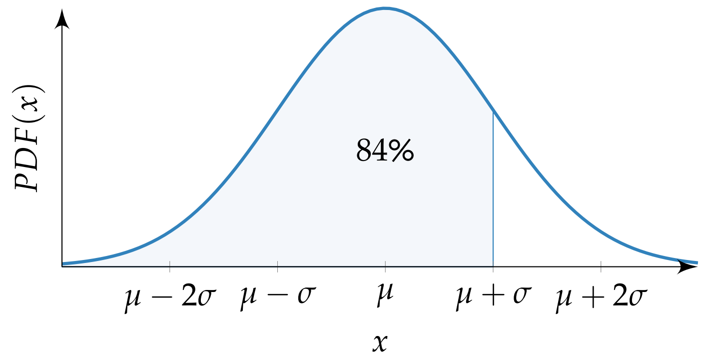


Figure 10.18: Conservative estimation of measured values from the environment perception modules

$\mu \pm \sigma$.[19] Such a one-sigma safety margin will cover $p = 84\,\%$ of all situations under the assumption of normally distributed state variables and the negligence of other uncertainty. Moreover, if the uncertainties in the perception modules decrease, the automated vehicle will become more agile in performing lane changes. In bad weather conditions, difficult domains, or if other factors increase uncertainty in the environment perception, the automated vehicle will become less adventurous to perform lane changes.

Lane Change Possible due to the Infrastructure Situation

To decide whether a lane change is possible, it is also necessary to consider the infrastructure. On the one hand it is vital to consider if the automated vehicle is currently driving on a valid lane and, on the other hand, if a neighbor lane exists and is valid. Moreover, it is mandatory to evaluate if lane markings, traffic signs, and traffic rules permit a lane change.

Past Implementations

The idea to evaluate the validity of the ego and neighbor lane as well as evaluating lane marking types as been demonstrated for instance by Kujawski (1995) and Schubert et al. (2010). Yet, both publications had a far more basic perception system and no map data to be merged in a context model. Thus their implementation were not able to address issues like traffic rules or how to handle lane changes in highway interchanges.

Valid Lanes

To check lane validity for lane changes, the lane attributes from the situation representation (cf. chapter 8) are evaluated. Only if the ego lane and the neighbor lane in the direction of the intended lane change has been confirmed by current sensor data and has steadily been tracked, they will be considered as valid lanes for performing a lane change.

Line Types

Traffic rules from traffic signs or line types are incorporated in an earlier step of scenery modeling as part of the context modeling. Lane change restrictions are translated into lane boundary type attributes derived from map and perception data. In case of incorrect maps, incorrect perception, or incorrect localization within that

[19] In theory, this conservative estimate is unnecessary given that all uncertainties are transformed as in section 10.3.4. Yet, this conservative estimate helps to remedy model insufficiencies and proved its use in real world driving.

maps these information are contradictory. For regular driving on a highway, the more lane change restrictive information is used. For lane changes in highway interchanges, the perceived information is used.

Last of all, it is evaluated, if the remaining length of a dashed line allowing lane changes is higher than the necessary length for performing a lane change. If not, no lane change will be started. To mitigate side effects of insufficient line type perception, a lane change will not be aborted due to a change of the line type, once it is in progress.

Ability Induced Skill Restrictions

Ability induced skill restrictions make a lane change impossible due to the general limitations of the vehicle. The concept of self representation as a foundation for planning and control has been demonstrated in Maurer (2000, p. 58 ff.), Bergmiller (2015, p. 145 ff.), and Reschka et al. (2015). Here the concept is applied towards lane change behavior planning.

Fast Neighbor Rear Vehicle on Streets Without Speed Limit

For instance, making a lane change to a left neighbor lane on a highway stretch without a speed limit may exceed the system's abilities because objects might approach the automated vehicle with extremely high velocities from behind. Attachment E calculates necessary sensor viewing ranges to the rear for fast approaching vehicles based on equation 10.6. Given the limited ability of the environment perception modules to track objects far ahead and a minimal time that is necessary to finish a lane change within, it may be that a fast vehicle from behind is forced to initiate a very strong emergency braking maneuver in order to prevent a collision with an automated vehicle which might be pursuing a lane change to the left neighbor lane. Simply because of the limited sensor viewing range to the rear, the automated vehicle's abilities might not allow lane changes to the left when there is no speed limit available. However, a lane change to the left on a highway without a speed limit will be possible if the environment perception modules see a slow vehicle on the neighbor lane, because a fast vehicle from behind will have to adapt its velocity to those vehicles anyway. Hence, starting a lane change to the left neighbor lane does not impose an unacceptable risk of collision.

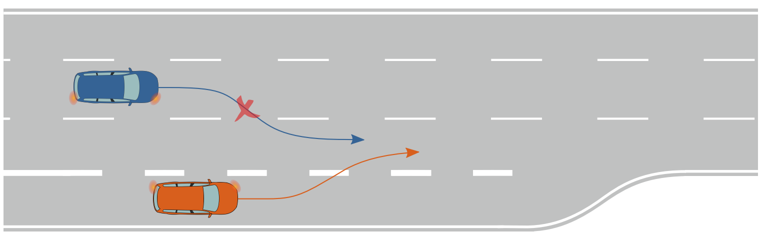


Figure 10.19: Avoiding collisions with merging vehicles due to perception ability limitations of an automated vehicle (blue)

Collision with Merging Vehicles

In some situations, general ability restrictions prevent the automated vehicle from lane changing. Figure 10.19 shows a situation where perception ability restrictions prevent the automated vehicle from performing a lane change to a right neighbor lane. Here, a highway on-ramp could result in a high risk of narrowly merging trucks with two lanes lateral offset not expecting a vehicle from another lane to change to the rightmost lane of a highway. Since perception abilities are insufficient to detect those vehicles and their merging intent on an on-ramp reliably in every situation, it is necessary to avoid such situations. In such a situation an automated vehicle should not initiate a lane change to a right neighbor lane possibly causing a risk of collision with those merging vehicles.

Ego Too Slow

A lane change may also be impossible, if the automated vehicle is driving too slowly. In those situations a lane change might take longer to complete than it is possible to predict environment changes reliably. Hence, another ability restriction for lane changes is imposed by the prediction abilities necessitated by the automated vehicle's absolute ego velocity.

Domain Type

Last of all, the environment domain type may impose ability restrictions for lane changes. For instance, currently the implementation is not able to offer safe overtaking maneuvers on country roads, where it is necessary to change to lanes with oncoming traffic. Other examples for lane change domain ability restrictions are areas with road works on highways.

Skill Restrictions

Skill restrictions rule lane changes impossible due to temporary issues. The concept of self representation as a foundation for planning and control has been demonstrated in Maurer (2000, p. 58 ff.), Bergmiller (2015, p. 145 ff.), and Reschka et al. (2015). Here the concept is applied towards lane change behavior planning. Among the skill restrictions being currently addressed are sensor viewing ranges to perceive objects and lanes.

Sensor Object Viewing Ranges

To calculate sensor viewing ranges, it is common to use occupancy grid based approaches (Elfes, 1987) and to apply it to free-space calculations as in Murray & Little (2000), Miura et al. (2002), or Badino et al. (2007) in automated driving. A challenge to these approaches is that they work well for static elements, but dynamic elements will be blurred in an occupancy grid. Hence, these approaches may only provide *one* input for sensor viewing range calculations. It needs to be supplemented by an approach which considers dynamic elements as well.

To obtain a sensor viewing range that can consider static *and* dynamic elements, a simple form of the incept theorem is used. Given another vehicle is driving directly behind or in front of an automated vehicle, it also obstructs the view[20] to the neighbor lanes. It may cause occlusions since the main sensors covering the rear and the front area are mounted at the center of the automated vehicle's front and rear bumper. The sensor viewing range for object detection on the neighbor lane can be calculated by the incept theorem:

$$s_{pos,\,viewing\ range} = tan\alpha \cdot s_{pos,\,obstruction} = \frac{0.5 \cdot w_{obstruction}}{0.5 \cdot (w_{ego\ lane} + w_{neighbor\ lane})} \cdot s_{pos,\,obstruction} \tag{10.10}$$

[20]This consideration holds true for sensors mounted in low viewing positions, e.g., on the bumpers. Other approaches might be viable for roof-mounted sensors or sensors mounted to automated trucks.

Figure 10.20 illustrates the incept theorem for a rear vehicle on the left neighbor lane and visualizes the parameters for equation 10.10.

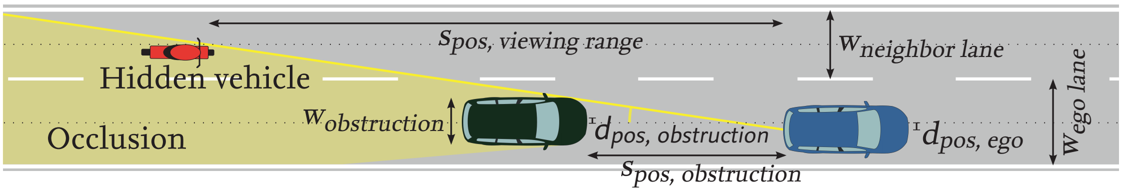


Figure 10.20: Limited sensor viewing range due to occlusions caused by a vehicle driving behind an automated vehicle (blue)

Simplifications

For the sake of simplicity, the curvature of the road is neglected. Additionally, the uncertainty prone lateral offset of the obstructing vehicle to its center of the lane ($d_{pos,\ obstruction}$) as well as the ego vehicle's offset ($d_{pos,\ ego}$) to the center of the lane (dashed line) are not considered. It is assumed that the first to be seen corner of a potentially hidden vehicle (red in Figure 10.20) is at least in the center of the neighbor lane. This assumption still seems reasonably valid for motorbikes on highways, but may be overly simple for bikes on urban roads or very curvy rural roads in a mountainous area.

Lane Viewing Ranges

To calculate lane viewing ranges for perceiving lanes, the ego lane segment and its neighbor lane segments perceived by the lane tracking algorithms are traversed to their rear end. If other lane segments were linked to those lane segments, the algorithm traverses through this graph until an end is found or a sufficient lane viewing length of currently 400 m is covered. The minimum of both is considered to be the lane viewing range to the rear. The same procedure is executed to the front. However, the viewing range to the front is much shorter since those locations have not yet been explored by the automated vehicle. Hence, viewing ranges to the front are typically not longer than 50 m or whatever a camera and the subsequent lane tracking algorithms are able to perceive and track. For calculating lane viewing ranges, the lane (segment) existence estimation itself is left to the lane information fusion module as part of the automated vehicle's perception system.

10.3.3 Gap Quality Assessment

Evaluating Several Gaps

Basic situation assessment for lane change decision making evaluates the current situation. However, a human driver is able to interact with the environment in order to make his situation more favorable for an intended maneuver such as a lane change. In dense traffic, this interaction enables a higher rate of success for the completion of an intended maneuver. It necessitates the evaluation of not only the current direct neighbor gap but also the overall environment of the vehicle. For this purpose, a numeric gap quality rating is calculated for every gap between dynamic traffic participants on the neighbor lanes.

Shortcomings of Existing Approaches

Section 6.2.2 reviewed models for gap acceptance and gap selection from the literature. While those models provide good results in simulation environments with perfect information, they are susceptible to inaccurate situation assessment as being based on acceleration calculations which is only the second derivative of the measured dynamic element positions. The higher the longitudinal distance of a gap is, the higher is the chance that their boundaries are prone to perception errors due to occlusions, false dynamic element associations or entirely missed vehicle tracks. Hence, a far simpler approach has been chosen, that is mostly based on distance calculations and required safety margins. Relative velocities and accelerations only need to be calculated towards an easier to perceive front vehicle of a gap and they can be weighted with a cost factor to balance gap quality information gain against inaccuracies in situation assessment due to measurement errors.

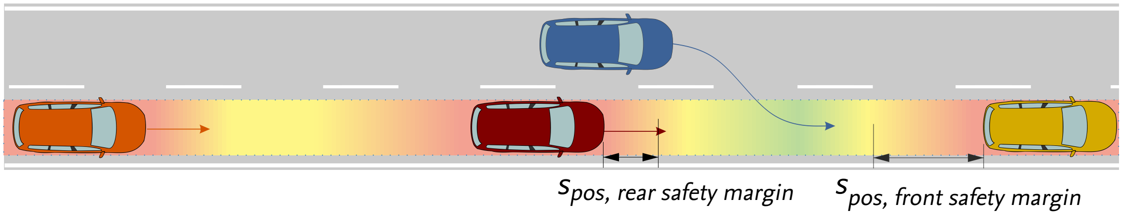


Figure 10.21: Gap quality assessment

Example

Figure 10.21 illustrates a typical driving situation on a multi-lane street in dense traffic. The traffic situation is based on a German highway with a right lane driving order. Several trucks moving at 80 km/h occupy the right lane, to which the automated vehicle shall change in order to take, e.g., the next exit. The automated vehicle (blue) is currently still driving faster (120 km/h) than the trucks on the neighbor lane.

Characteristics of a Gap

The gap quality is assessed by the geometric distance, length, and relative velocity of the gap towards the ego vehicle. Every gap is defined by a front and rear vehicle, each with a relative position, velocity, and acceleration. Even if no vehicles are detected, the sensor viewing range may define potential virtual objects to describe a gap.[21] Within each gap a target pose exists as the best possible pose to reach within this gap. Such a target pose may not only be a singular point but also a multi-dimensional interval of position, velocity, and acceleration ranges. It is the task of a trajectory planning module to find the single – to an optimization criteria – best trajectory within these intervals.

Trajectory Planning Heuristics

To avoid running a trajectory planning module for several gaps in several future time steps to find the best gap over a given time horizon, a heuristic model for the trajectory planning is used. First, the necessary distance towards the front and the rear vehicle defining a gap is calculated by the equations in section 10.3.2. The relative target pose for the gap is allowed to vary in the interval between the front and the rear vehicle minus these safety distance margins. As a goal target pose, approximately 5% further in front than in the center of the gap proved to be a good target pose in driving tests performed by the author ($\lambda = 0.55$). As a gap velocity

[21] Such a virtual object is placed with a longitudinal distance of $s_{dist} = 100$ m in front or behind the last seen object with a velocity equal to the speed limit of that particular lane where it is placed. Assuming that there will not be a static element right in front of the viewing range of the automated vehicle is a common, yet dangerous assumption to enable automated driving with velocities typically driven on highways. However, this issue still holds true for objects in the ego lane and is thus independent of lane changing.

and acceleration, the velocity and acceleration of the front vehicle is used, as this will become the relevant longitudinal control target after changing into the gap.

This yields the following formula for a target pose in a gap:

$$\vec{s}_{gap\,tp} = \begin{pmatrix} pos \\ vel \\ acc \end{pmatrix} = \begin{pmatrix} s_{pos,\,gap\,tp} \\ s_{vel,\,gap\,tp} \\ s_{acc,\,gap\,tp} \end{pmatrix} = \begin{pmatrix} \lambda s_{pos,frontObj} - (1-\lambda) s_{pos,\,rearObj} \\ s_{vel,frontObj} \\ s_{acc,frontObj} \end{pmatrix} \tag{10.11}$$

For the trajectory planning implementation in the test vehicle (cf. attachment B) it proved sufficient to specify a position sampling interval only. This sampling makes it possible to target any position within a gap, which is shortened by a front and rear safety margin ($s_{pos,front\,safety\,margin}$, $s_{pos,\,rear\,safety\,margin}$) towards the front and rear vehicle of that gap.

Sampling

If the automated vehicle ($s_{pos,\,ego}$) is behind the rear gap safety interval (first case of equation 10.12) the delta target pose within that gap is set to the closest point of that gap, directly at the beginning of the rear gap safety margin. If the automated vehicle is in front of the target gap, the delta target gap pose is set to the front end of the gap directly bounded by the front gap safety interval ($s_{pos,front\,safety\,margin}$) towards the front vehicle (last case of equation 10.12). If the automated vehicle is within the front and rear gap safety intervals, the delta pose is set to the position offset between the automated vehicle and the earlier calculated $s_{gap\,tp}$. The delta gap pose $\Delta\vec{s}_{gap\,tp\,safe}$ is a measure for how much position, velocity, and acceleration adaption is necessary to adjust to a gap.

$$\Delta s_{pos,\,gap\,tp\,safe} = \begin{cases} s_{pos,\,gap\,tp} - s_{pos,\,ego} + s^{-}_{pos,\,gap\,interval} \\ \qquad \text{if } s_{pos,\,ego} - s_{pos,\,gap\,tp} < s^{-}_{pos,\,gap\,interval} \\ \\ s_{pos,\,gap\,tp} - s_{pos,\,ego} \\ \qquad \text{if } s^{-}_{pos,\,gap\,interval} \leq s_{pos,\,ego} - s_{pos,\,gap\,tp} \leq s^{+}_{pos,\,gap\,interval} \\ \\ s_{pos,\,gap\,tp} - s_{pos,\,ego} + s^{+}_{pos,\,gap\,interval} \\ \qquad \text{else} \end{cases} \tag{10.12}$$

$$\Delta\vec{s}_{gap\,tp\,safe} = \begin{pmatrix} \Delta s_{pos,\,gap\,tp\,safe} \\ s_{vel,\,gap\,tp} - s_{vel,\,ego} \\ s_{acc,\,gap\,tp} - s_{acc,\,ego} \end{pmatrix} \tag{10.13}$$

The front and rear safety margins are calculated based on the same motion equations used for the lane change possible assessment in section 10.3.2 in equation 10.6. As a braking acceleration $s_{acc,RN} = -1\,\mathrm{m/s^2}$ or $s_{acc,FN} = -1\,\mathrm{m/s^2}$ is used to emulate a driver cooperatively opening a gap (cf. section 10.3.2) when no lane change has been initiated so far. The desired time gap T_{RN} and reaction time T_R are parameterized accordingly as in section 10.3.2.

Safety Margins

$$s^{+}_{pos,\,gap\,interval} = max(s_{pos,frontObj} - s_{pos,front\,safety\,margin} - s_{pos,\,gap\,tp}, 0.0) \tag{10.14}$$

$$s^{-}_{pos,\,gap\,interval} = -max(s_{pos,\,gap\,tp} - s_{pos,\,rear\,safety\,margin} - s_{pos,\,rearObj}, 0.0) \tag{10.15}$$

Costs of Gap Adjustment

Every ego-relative deviation from a gap target pose $\Delta\vec{s}_{gap\,tp\,safe}$ is penalized by a cost term.[22] If a safety margin towards either a front or rear object is violated, it is penalized by a cost term. The first minuend in equation 10.16 reflects costs for absolute position, velocity, and acceleration offsets between a gap and the automated vehicle $\vec{c}_{gapAdj} = (c_{gapAdj,\,pos}, c_{gapAdj,\,vel}, c_{gapAdj,\,acc})^t$. The second and third minuend in equation 10.16 penalize the violation of the safety margins.

$$\begin{aligned} gap\,quality &= 1 \quad -\vec{c}_{gapAdj} \cdot \left|\Delta\vec{s}_{gap\,tp\,safe}\right|_{element\,wise} \\ &\qquad -c_{front\,safety\,margin} \cdot max(s_{pos,frontObj} - s_{pos,\,gap\,tp}, 0) \\ &\qquad\qquad \cdot\Theta(s^{+}_{pos,\,gap\,interval}) \\ &\qquad -c_{rear\,safety\,margin} \cdot max(s_{pos,\,gap\,tp} - s_{pos,\,rearObj}, 0) \\ &\qquad\qquad \cdot\Theta(-s^{-}_{pos,\,gap\,interval}) \end{aligned} \tag{10.16}$$

The gap quality is calculated for a gap directly next to the automated vehicle and two gaps to the front and two gaps to the rear to the left and right respectively. Thus, in total 10 gaps around the automated vehicle are evaluated in every situation assessment cycle.

10.3.4 Estimating and Propagating Uncertainties [23]

Among the key challenges for tactical lane change behavior planning is the inherent uncertainty (cf. section 2.5) from any kind of environment perception modules and the nondeterministic behavior of the world itself. The higher the abstraction level of the perception becomes, the bigger the uncertainty about state estimates (velocities, intentions, etc.) will get. For instance, reliably estimating the velocity or acceleration of an element is already difficult but estimating the intention to merge or to let

[22] In fact, the quality assessment could be considered as part of the cost and reward model rather than the situation assessment model. However, cost-weighted gap qualities proved to be more stable than their individual constituting factors. Thus, calculating overall gap qualities in the situation assessment proved to be simpler than performing it in the reward model only. Moreover, it allowed time domain filtering in the dynamic Bayesian network.

[23] Part of this subchapter has been pre-published by the author in Ulbrich & Maurer (2015a).

someone merge is even harder. At last, nondeterministic behavior resulting from disturbances to the system (cf. moose in section 2.5) will always cause a remaining uncertainty.

Tackling Uncertainties

As illustrated in Figure 10.9, every hidden state variable of the dynamic Bayesian network will be estimated based on old state information and current measurement updates. Every single one of these measurement updates will come with an uncertainty.[24] For instance, all distances or velocities of objects will not only have an expected value μ from the Kalman filters in the environment perception modules but also a variance σ^2. As the expected values μ of specific measurement updates propagate to some hidden state variable estimates, so will their variances according to the measurement update model explained so far in this section. Non-linear inter-dependencies between state variables cause a non-linear propagation of those variances through the dynamic Bayesian network to represent beliefs about the driving situation. An approach to address this is to use an unscented transform with a minimal set of sigma points in the same way it is used in an unscented Kalman filter (Thrun et al., 2005, p. 65).

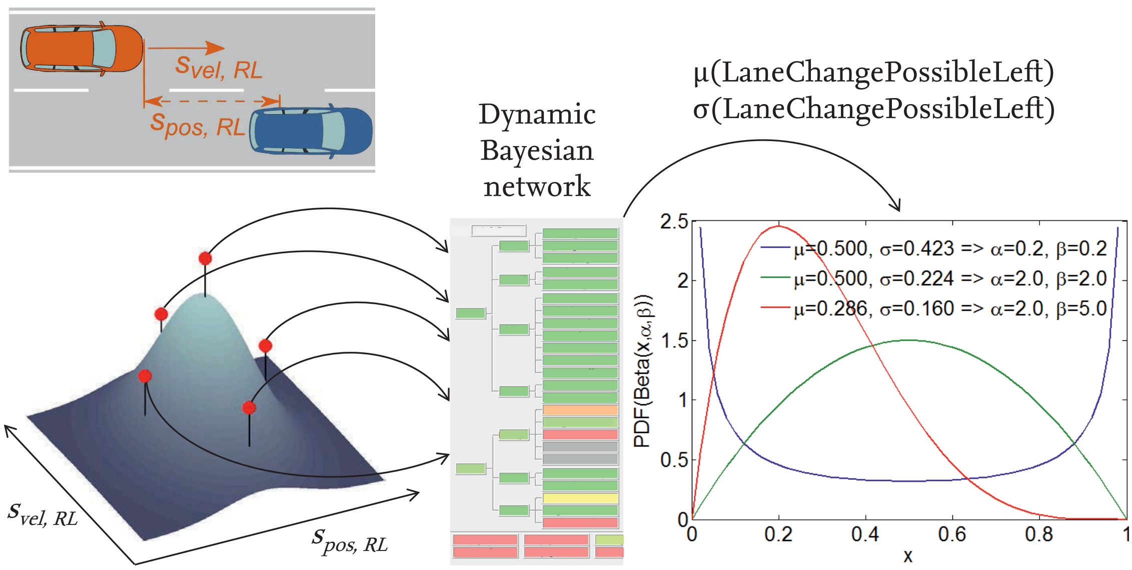


Figure 10.22: Sigma point variance propagation

Unscented Transform

The unscented transform is a method to calculate the statistics of random variables through a non-linear transformation function like a Bayesian network $g(z, u)$. It uses a heuristic approach to propagate uncertainties in measured state variables into uncertainties of derived variables along n dimensions. The input variables are modified such that $L = 2 \cdot n + 1$ sigma points Z_i are selected by not only using the

[24]According to section 2.5, the approach here is used to address the measurement uncertainty. The existence uncertainty is currently only considered in the lane change beneficial estimation due to its current performance limitations (cf. section 10.3.5). The association uncertainty is currently not considered. Yet, they are indirectly considered by their effect on the measurement uncertainty.

means of all variables $\bar{z}$, but also by adding and subtracting the weighted covariances Σ of these:

$$Z^{[0]} = \bar{z} \tag{10.17}$$

$$Z^{[i]} = \bar{z} + \left(\sqrt{(n+\lambda)\cdot\Sigma}\right)_i \qquad \forall\, i = 1, ..., n \tag{10.18}$$

$$Z^{[i]} = \bar{z} - \left(\sqrt{(n+\lambda)\cdot\Sigma}\right)_{i-n} \qquad \forall\, i = n+1, ..., 2n \tag{10.19}$$

$\lambda = \alpha^2 \cdot (n + \kappa) - n$ is calculated from the scaling parameters α and κ. The matrix of the sigma points $Z^{[i]}$ is transformed by the given non-linear transfer function $g(z, u, b_{t-1})$ (the Bayesian network, cf. section 10.3), and thereby changes the shape of the initial probability distributions:

$$X^{[i]} = g\left(Z^{[i]}, u, b_{t-1}\right) \tag{10.20}$$

The transformed sigma points $X^{[i]}$ do follow a transformed random distribution with mean μ' and covariance Σ'.

$$\mu' = \sum_{i=0}^{2n} w_m^{[i]} \cdot X^{[i]} \tag{10.21}$$

$$\Sigma' = \sum_{i=0}^{2n} w_c^{[i]} \cdot (X^{[i]} - \mu')(X^{[i]} - \mu')^T \tag{10.22}$$

Using the weighting factors $w_c^{[i]}$ and $w_m^{[i]}$:

$$w_m^{[0]} = \frac{\lambda}{n+\lambda} \tag{10.23}$$

$$w_c^{[0]} = \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \tag{10.24}$$

$$w_c^{[i]} = w_m^{[i]} = \frac{1}{2\cdot(n+\lambda)} \qquad \forall i = 1, ..., 2n \tag{10.25}$$

Applying an Unscented Transform

Behavior planning for lane changes necessitates a high dimensional measurement and state space. Thus, the number of sigma points $L = 2 \cdot n + 1$ would be very high if every dimension is varied. Thus, it is beneficial to identify which dimensions result in the most significant variances for the derived decision relevant variables. For the issue of lane change decision making, the position and velocity of the objects closest in every region of interest (cf. Figure 10.15) as well as the ego velocity are most relevant. For five regions of interest (ROI) in Figure 10.15 and the ego velocity this results in $n = 11 = 5 \cdot 2 + 1$ dimensions for the unscented transform and therefore $L = 2 \cdot n + 1 = 23$ sigma points.[25]

[25] Accelerations are ignored due to being too noisy.

For the application of lane change decision making, it is assumed that the covariance matrices Σ and Σ' are just holding non-zero elements σ_m^2 on the principal diagonal. Depending on the type of resulting (hidden) state variables of the Bayesian network, their mean and variances are either interpreted as the mean and variance of a normal distribution for distances, velocities, accelerations, and beta distributions for probability estimates.

A beta distribution is a value-continuous probability distribution for random variables. Other than a normal distribution it is restricted to the range $[0,1]$. It is described by two shape parameters α and β. Those parameters allow several skewed probability distributions to be modeled as depicted in Figure 10.22. The mean and variance are defined by:

Beta Distribution

$$E(x) = \frac{\alpha}{\alpha+\beta} \tag{10.26}$$

$$Var(x) = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{10.27}$$

Solving these equations for α and β as a function of the mean and variance derived from the unscented transform yields:

$$\alpha = E(x) \cdot \left(\frac{E(x)\cdot(1-E(x))}{Var(x)} - 1 \right) \tag{10.28}$$

$$\beta = (1-E(x)) \cdot \left(\frac{E(x)\cdot(1-E(x))}{Var(x)} - 1 \right) \tag{10.29}$$

Figure 10.23 illustrates an estimated mean and variance for a beta distributed random variable to represent the probability that a gap is suitable for making a lane change.

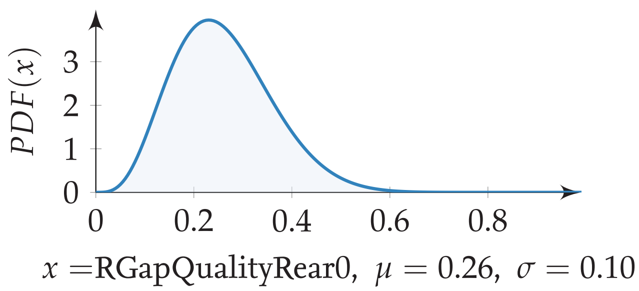


Figure 10.23: Example of a hidden state variable as an estimate for a gap quality. The variance is derived from measured variances of object positions, velocities, and accelerations

All the uncertainties discussed so far are considered *within* a time slice. However, the perceived environment might also change *between* two time slices. E.g., by perceiving an additional object or losing a formerly tracked object, the abstract hidden state variables, if – for instance – a lane change is possible, have to encompass more uncertainty than just that resulting from pure position, velocity, and acceleration variances only. These additional uncertainties will result in changes of the hidden

Autoregressive Models

state variables over time. In time series analysis, this residual changes in a time series due to an innovation process are modeled by autoregressive (AR) models. The short-run residual changes of a random process's variance over time is called conditional heteroscedasticity (Enders, 2015, p. 119).

In an autoregressive model of a random process $\{x_t\}$ the value of a random variable is given by the last value plus the residual change $\epsilon_t = x_t - x_{t-1}$, where $\{\epsilon_t\}$ is assumed as a white noise process with variance σ^2. The time series can be modeled by $x_t = a_0 + \sum_{i=1}^{L} a_i \cdot x_{t-1} + \epsilon_t$, with $E(x_t) = \sum_{i=1}^{L} a_i \cdot x_{t-1}$ (Enders, 2015, p. 50 ff.).

Time-Varying Variances

An autoregressive conditional heteroscedasticity model (ARCH) or its generalization (GARCH) are used to model if the variance σ^2 is time-dependent (cf. Enders, 2015, p. 129 ff. or Barber, 2012, p. 526). Here, the variance of the residual change $Var(\epsilon_t) = E[(x_t - \mu)^2]$ measures the variability (or in financial science the *volatility*) of the random process: $\sigma_t^2 = \alpha_0 + \sum_{i=1}^{Q} \alpha_i \cdot (x_{t-i} - E(x_{t-i}))^2 + \sum_{i=1}^{P} \beta_i \cdot \sigma_{t-i}^2$ With $P = Q = 1$ this broils down to $Var_{innovation}(x_t) = \sigma_t^2 = \alpha_0 + \alpha_1 \cdot (x_{t-1} - E(x_{t-1}))^2 + \beta_1 \cdot \sigma_{t-1}^2$.

The total variance is calculated by:

$$\begin{aligned} Var(x_n) =&(1 - w_n) \cdot Var(x_{n-1}) \\ &+ w_n \cdot (Var_{measure}(x_n) + Var_{innovation}(x_n)) \end{aligned} \tag{10.30}$$

With a perfect perception system, which estimates all perception uncertainties correctly, the measurement variance would reflect all perception uncertainty. For a real world perception with incomplete and imperfect uncertainty estimation, the innovation variance and a by w_n weighted average of current and former variances help to alleviate these imperfections using application-specific knowledge about how much the variances of certain aspects are supposed to change in a certain time interval.

Limitation

The reader should note that the here presented approach only addresses the *measurement* uncertainty as part of the *perception* uncertainty; not the *execution* uncertainty and the *prediction* uncertainty introduced in section 2.5. For a research project, this simplification may to some extend be justified because it ensures that the system works at least *most of the time*. Rare events like a moose running on the street, or a broken steering system are currently left to the responsibility of a safety driver. For a market-ready SAE level three to five system (cf. section 2.1) this will not be acceptable and opens a wide field of future research.

10.3.5 Consistency for Measurement Belief Updates

Section 5.2.1 and Figure 10.9 introduced the concept of a dynamic Bayesian network. So far, it has been discussed how to aggregate information about distances and velocities into more aggregated hidden state variables to represent if a lane change is possible or beneficial and the suitability of different gaps for lane changes. Hence, this chapter focused on the "bottom-up" aspects of a dynamic Bayesian network as in Figure 10.9.

Yet, a dynamic Bayesian network not only entails this dependency *within* a time slice, but also the state dependencies *among* time slices. This is illustrated by the horizontal arrows in Figure 10.9. Based on the Markov assumption (cf. section 5.2.1), the state variables of the current time slice only depend on the random variables of the *immediate last* time slice. As predictions of hidden random variables from the last time slice are challenging, new state estimates are calculated simply by a first order low-pass filter with a weighting factor/filter time constant w_n from the previous state estimate $E(x_{n-1})$ and a new, measurement-based state estimate:

Temporal Filtering

$$E(x_n) = (1 - w_n) \cdot E(x_{n-1}) + w_n \cdot (E_{measure}(x_n)) \tag{10.31}$$

The weighting factors w_n are used to tailor an application-specific trade-off between consistency and responsiveness for each hidden state variable.

So far, this low pass filtering only ensures a certain degree of consistency for hidden state variables. Inaccurate object velocities, positions, and existence estimates still compromise state estimates to a certain degree. However, it is possible to improve state estimates by tailoring state variables in a situation description specifically to the needs of a driving function regarding a consistency-responsiveness trade-off.

Tailoring

High-level knowledge can be incorporated into a situation description to make it more useful for a particular driving function. For instance, objects are unlikely to suddenly disappear from a particular lane or towards the direct left and right proximity of the automated vehicle, where only radar sensors may be able to detect reflections from passing vehicles. Moreover, object movements can be stabilized by incorporating information about lanes. Objects are assumed to move along lanes.

High-level Knowledge

To achieve this, a second, simple, *application-specific*, multi object tracking is implemented for the situation interface and as part of the measurement model to obtain better function-specific state belief estimates. The object tracking is implemented by instantiating a Kalman filter (cf. section 5.2.1) for each track. The movement in longitudinal direction towards a lane is assumed to be independent of its lateral movement perpendicular to a lane. In longitudinal, as well as in lateral direction, a simple point-of-mass movement model is used. As a simplifying assumption, objects are assumed to only move within a lane. Hence, new measurements of a scene are associated with tracks based on the distance metric:

High-level Object Tracking

$$dist = \frac{0.6}{\mathrm{m}} \cdot \left|s_{pos,track} - s_{pos,meas}\right| + \frac{0.4}{\mathrm{m/s}} \cdot \left|s_{vel,track} - s_{vel,meas}\right| \tag{10.32}$$

A measurement is associated to a track as long as the distance *dist* is smaller than $\epsilon = 2.0$. If the distance is higher, a new track will be instantiated. A track will be abandoned if it is not updated for more than 0.3 seconds. A track will not be propagated to the situation assessment if it is not updated for more than 0.1 seconds or has not been tracked for at least 0.1 seconds.[26]

[26] Based on the track age, a dynamic element existence uncertainty is estimated. This proved to be more stable than the one derived from the central sensor data fusion module.

The implementation details for such a high-level tracking are omitted since object tracking is not the main focus of this thesis.

10.4 Situation Prediction Model

The situation prediction model facilitates a prediction of the entire situation as a function of the former situation and a decided action $b_{t+\tau} = f(b_t, z_t, u_t, \tau)$. The state belief distribution $b_{t+\tau}$ entails hidden and observable state dimensions. It is far simpler to predict the observable state components, like distances, velocities, and accelerations in a particular situation and subsequently to calculate the thereof resulting changes to the hidden state variables. To predict observable situation aspects, well-established behavior prediction models can be used. To predict the impact on hidden state variables, the measurement model from section 10.3 is used.

Situation Prediction

The situation prediction as a whole is illustrated in Figure 10.24. First of all, the ego movement is predicted using the intelligent driver model (cf. section 10.4.1) and a lane following assumption (cf. section 10.4.2). Based on the ego movement, the changes in the dynamic environment are predicted in all relevant (cf. section 10.3.2) regions of interest (ROI). For this, the same prediction model as in sections 10.4.1 and 10.4.2 are used. The only difference is that the predicted ego movement must be compensated according to the coordinate system definition in attachment D and a transition of objects between region of interests must be ensured if a vehicle on the neighbor lane falls behind the ego vehicle or is predicted to overtake it. Further the lane change status transition is predicted according to the action u_t as in Figure 10.5. If a lane change was in progress and is predicted to be finished by *FinishLcLeft/Right* a transition of the ego vehicle to the neighbor lane is incorporated in the prediction. For this, a switch of the lane elements to the neighbor lane element is performed and all dynamic elements are moved to the new resulting region of interest. Last of all, the hidden state variables of a state belief distribution are updated.

10.4.1 Longitudinal Driver Model

For longitudinal prediction of any dynamic element in the scene, a so-called driver model (cf. section 6.2.1) is used. In this thesis, an augmented version of the so-called "enhanced intelligent driver model" (cf. Kesting et al., 2010) based on Treiber & Helbing's (2001) "intelligent driver model" is used. This augmented version has been refined by Shen & Jin (2012). It is used because of its low computational demands and capability to model interactions between vehicles. Because it is based on state variables (position, velocity, and acceleration in longitudinal/lateral direction) in can make use of parametric uncertainty representations estimated for those variables in a Kalman filter based tracking in environment perception.

An update of a dynamic element's longitudinal position s_{pos} and its velocity s_{vel} is calculated based on a calculated acceleration $s_{acc,final}$:

$$s_{vel}(t + \Delta t) = s_{vel}(t) + s_{acc,final}(t + \Delta t) \cdot \Delta t \quad (10.33)$$

$$s_{pos}(t + \Delta t) = s_{pos}(t) + s_{vel}(t + \Delta t) \cdot \Delta t \quad (10.34)$$

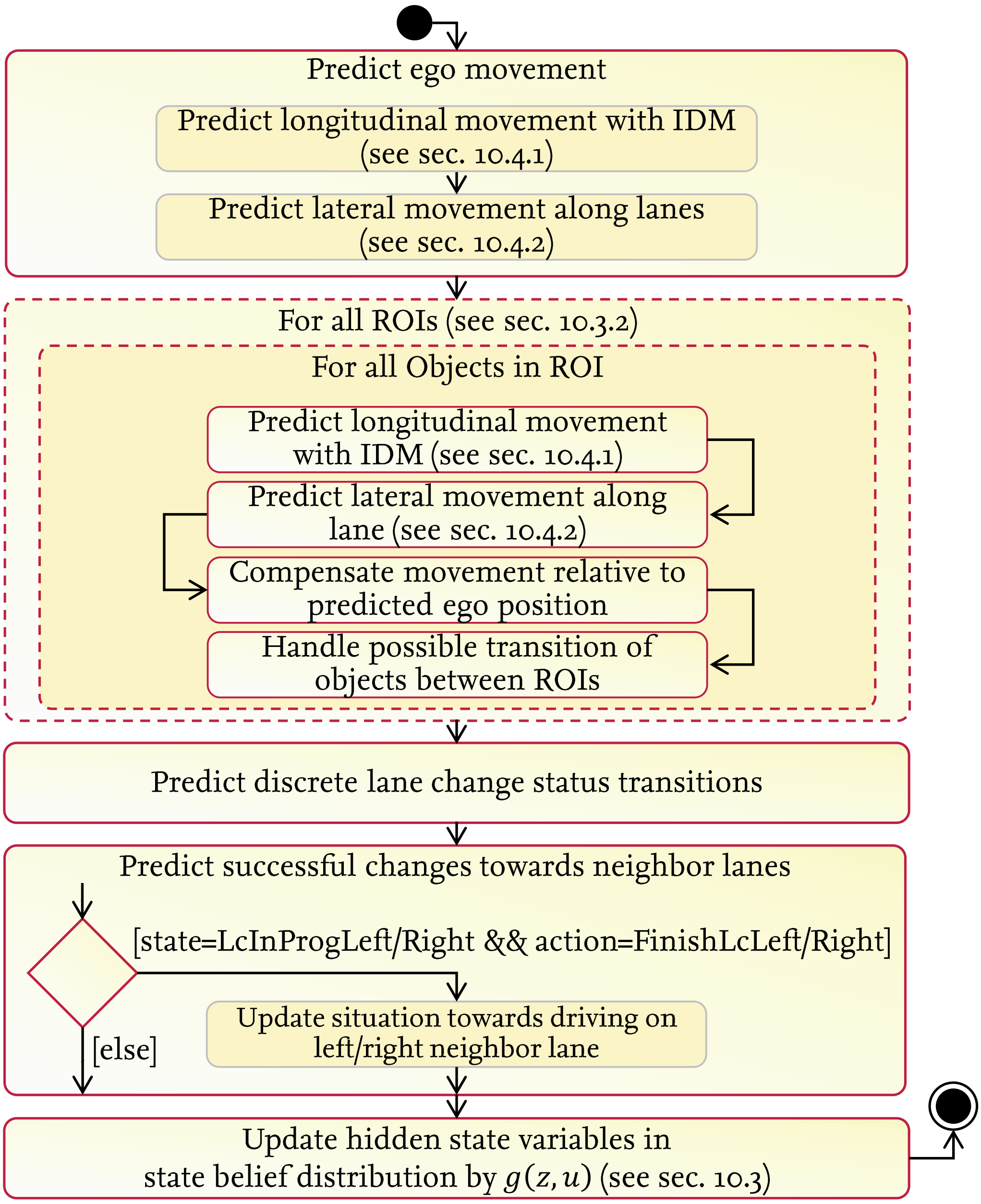


Figure 10.24: Simplified illustration of the situation prediction model $b_{t+\tau} = f(b_t, z_t, u_t, \tau)$

The necessary acceleration $s_{acc,final}$ is calculated by:

$$s_{acc,final} = \begin{cases} s_{acc,idm} & \text{if } s_{acc,idm} \geq s_{acc,cah} \\ (1-c) \cdot s_{acc,idm} + c \cdot s_{acc,adjust} & \text{else} \end{cases} \tag{10.35}$$

Enhanced Intelligent Driver Model

The necessary acceleration is calculated by the acceleration of Treiber & Helbing's (2001) "intelligent driver model" as long as this acceleration $s_{acc,idm}$ is higher than the constant-acceleration-heuristic-based acceleration introduced in the "enhanced intelligent driver model" by Kesting et al. (2010). This heuristic helps to avoid overreactions of the initial "intelligent driver model" with low relative velocities and smaller front gaps than the desired gap length. This is particularly relevant for cut-in scenarios and lane changes. This heuristic of the "enhanced intelligent driver model" is based on the assumption of human drivers that a front vehicle will not suddenly initiate an emergency braking maneuver without any reason and that it is thus acceptable to slowly increase the front gap instead of a braking maneuver with the maximal possible braking deceleration. The "constant-acceleration heuristic" takes into account the front vehicle's acceleration. By $\tilde{a}_{FE} = min(s_{acc,FE}, s_{acc,max})$ it is ensured that the resulting acceleration remains below $s_{acc,max}$.

If the "intelligent driver model" results in a higher deceleration than the "constant-acceleration heuristic", a weighted mixture of both is used with c as a "coolness weighting factor".

The variables $s_{acc,cah}$ and $s_{acc,adjust}$ are calculated by:

$$s_{acc,adjust} = s_{acc,cah} + b_{com} \cdot tanh\left(\frac{s_{acc,idm} - s_{acc,cah}}{b_{com}}\right) \tag{10.36}$$

$$s_{acc,cah} = \begin{cases} \frac{s_{vel}{}^2 \cdot \tilde{a}_{FE}}{s_{pos}{}^2_{vel,FE} - 2 \cdot s \cdot \tilde{a}_{FE}} & \text{if } s_{vel\,FE} \cdot \Delta s_{vel} \leq -2 \cdot s_{pos} \cdot \tilde{a}_{FE} \\ \tilde{a}_{FE} - \frac{\Delta s_{acc} \cdot \Theta(\Delta s_{vel})}{2 \cdot s_{pos}} & \text{else} \end{cases} \tag{10.37}$$

$$\tilde{a}_{FE} = min(s_{acc,FE}, s_{acc,max}) \tag{10.38}$$

Extension by Shen et al.

Shen & Jin (2012) once again extended the "enhanced intelligent driver model" by adding the governing control part $(\Theta(\Delta s_{vel}) + \Theta(s_{pos,b} - s_{pos}))$. This was initially developed to improve acceleration behavior when a signal turns green at a signalized intersection but it is also useful in stop and go traffic jam situations. By this, the deceleration part only kicks in if the dynamic element is approaching its front vehicle $(\Delta s_{vel} \geq 0)$ or if the front gap is smaller than a distance threshold $s_{pos,b} = h \cdot s^*_{pos}$. This is achieved by employing the step function Θ.

The acceleration from the augmented extended intelligent driver model is calculated by:

$$
\begin{aligned}
s_{acc,idm} &= s_{acc,free} \qquad\qquad\qquad\qquad + s_{acc,brake} \\
&= \overbrace{s_{acc,max}\left[1-\left(\frac{s_{vel}}{s_{vel,d}}\right)^{\delta}\right]}^{\textit{free acceleration term}} \overbrace{+\, b_{com}\left(\frac{s^*_{pos}}{s_{pos}}\right)^{2}\left(\Theta(\Delta s_{pos})+\Theta(s_{pos,b}-s_{pos})\right)}^{\textit{deceleration term to keep driving safe}}
\end{aligned}
\tag{10.39}
$$

Intelligent Driver Model

The intelligent driver model combines the free-road acceleration with a necessary deceleration if the gap in front of the dynamic element towards its front element is not larger than the “desired safety gap” s^*_{pos}. With:

$$s^*_{pos} = s_{pos,min} + s_{vel}T + \frac{s_{vel}\Delta s_{vel}}{2\sqrt{s_{acc,max}\cdot b_{com}}} \tag{10.40}$$

$$\Delta s_{vel} = s_{vel} - s_{vel,FE} \tag{10.41}$$

Desired Safety Gap

The “desired safety gap” s^*_{pos} is calculated as the sum of the, for low velocities relevant, minimum safety distance towards a front vehicle s_{min}, the distance corresponding to the “desired time gap” T and a third component to implement the “intelligent driving behavior”. This last component limits the braking deceleration in normal situations to a “comfortable braking deceleration” b_{com}. However, if stronger braking is necessary, it will likewise make the “intelligent driver model” collision free (Treiber & Helbing, 2001).

Parameters

The parameters in Table 10.1 were used according to best practices in Shen & Jin (2012), Kesting et al. (2010), and Treiber & Helbing (2001).

Table 10.1: Parameters for longitudinal driving behavior model

Parameter	Value	Description
$s_{vel,d}$	e.g. 36 m/s	Desired target velocity from scene
$s_{pos,min}$	2.5 m	Minimum safe distance
$s_{acc,max}$	3 m/s^2	Maximal acceleration
b_{com}	2.2 m/s^2	Comfortable braking deceleration
T	1 s	Driver’s reaction time
δ	4	Free acceleration exponent
h	1	Safe gap length multiplier
c	0.99	Weighting factor

Simplification

This model is used to predict the movement of every dynamic element on the lanes in the situation (cf. section 8.2). This entails the ego vehicle as well as all other vehicles in the environment of the ego vehicle. In this thesis, the predictions are limited to predictions in the ego lane. Lane changes of other vehicles are not predicted in the situation prediction itself. This turns out to keep predictions under uncertainty more structured. This simplification is compensated by the heuristic for fast rear vehicles in the lane change possible evaluation in section 10.3.2.

10.4.2 Lateral Prediction

Maintain Lateral Distance

For this thesis, it was sufficient to keep the lateral prediction model very simple. Every dynamic element is predicted to maintain its lateral distance to the center of its lane $d_{pos}(t + \Delta t) = d_{pos}(t)$. The lateral velocities and accelerations are assumed to converge to zero over a prediction horizon $d_{vel}(t + \Delta t) = 0$ and $d_{acc}(t + \Delta t) = 0$.

10.4.3 Handling Discontinuities in Situation Prediction

Handling Discontinuities

Planned actions may change the situation itself. For instance, if the abortion of a lane change is commanded to the prediction model, it will change the prediction of the ego vehicle as well as that for the other elements of the situation. Likewise, switching on the indicator or starting a lane change will change the situation prediction. These discontinuous changes in the situation prediction are handled by a system state transition matrix as in section 10.2.1.

10.5 Reward and Cost Model

The reward model provides numeric rewards and costs for executing a specific action in a particular situation. The rewards of a sequence of actions are aggregated to obtain an overall reward. The – to an optimization criteria – best sequence of actions will be selected.

Reward Dimensions

In a typical partially observable Markov decision process (POMDP) or a model predictive control (MPC) problem, the reward is often a scalar, numeric value (cf. sections 5.2.2 and 5.2.3). For the context of lane change planning this may result in odd behavior if a future, high lane change beneficial estimate could possibly cancel out a currently low lane change possible reward. Hence, the author suggests using a vector of rewards to host different reward dimensions as in equation 10.43. By this, reward dimensions can be aggregated separately over time and then thereafter be aggregated into an overall reward. In a classical POMDP it is a scalar. As such, the tradeoff between different reward dimensions would already be made within a time slice. The aggregated reward is summarized to a scalar by a modified version of equation 5.3:

$$R_T = min\left(\sum_{t_n=0}^{T} \gamma^{t_n} r_{LcPossible}(b(t_n), u(t_n)), \sum_{t_n=0}^{T} \gamma^{t_n} r_{LcBeneficial}(b(t_n), u(t_n))\right) \tag{10.42}$$

The reward dimensions of the reward vector $\vec{r}(b(t_n), u(t_n))$ are whether a lane change is beneficial (cf. section 10.3.1), whether a lane change is possible (cf. section

10.3.2), and gap qualities (cf. section 10.3.3) for every considered gap around the automated vehicle.

$$\vec{r}(b(t_n), u(t_n)) = \begin{pmatrix} r_{LcPossible}(b(t_n), u(t_n)) \\ r_{LcBeneficial}(b(t_n), u(t_n)) \\ r_{LeftGapQuality0}(b(t_n), u(t_n)) \\ \dots \\ r_{LeftGapQuality4}(b(t_n), u(t_n)) \\ r_{RightGapQuality0}(b(t_n), u(t_n)) \\ \dots \\ r_{RightGapQuality4}(b(t_n), u(t_n)) \end{pmatrix} \tag{10.43}$$

The reward dimensions for gap qualities are used to influence the selection of the best gap over a planning horizon in the action selection. To improve computational efficiency, not all gaps are considered as part of the possible action alternatives $u(t_n)$ but only the most promising gap. Thus, scenarios where the best possible gap changes while planning ahead are intentionally neglected. By adjusting towards a best possible gap, the gap quality implicitly becomes part of the the lane change possible/beneficial evaluation.

The reward model in Figure 10.25 assigns costs to each discrete action in each discrete lane change state (cf. section 10.2.1) as part of the overall system state vector. The reward is calculated depending on lane change state and action combinations framed by calculations for preparing the inputs from the state belief and post processing steps.

Further Details

If the reader wants to re-implement the algorithms, attachment G provides further details for the calculation of $\vec{r}(b(t_n), u(t_n))$. Unfortunately, the reward dimensions cannot easily be expressed by simple mathematical equations. At its core, the reward evaluation function does a case distinction for assigning different rewards towards executing particular actions in different lane change states. Further, it implements a hysteresis for initiating maneuvers and compensates numeric rewards from just transitioning into executing a maneuver. In attachment G, Algorithm 3 shows the main reward function. It utilizes state and action specific sub-functions 4, 6, 8, helper functions 9, 10, 11, 12, and the parameters in Table G.1.

10.5.1 Consideration of Cooperation [27]

Addressable Scenarios

As introduced in section 4, cooperative behavior is limited to very few situations and maneuvers due to the lack of appropriate communication and awareness channels for today's automated vehicles. For this thesis, the focus is limited to cooperation without explicit Vehicle-To-X communication. Any approaches that necessitate – perhaps one day existing – broad availability of Vehicle-To-X communication in all road vehicles will not be considered. At the other end of the axis of the commu-

[27]This section has been pre-published by the author in Ulbrich et al. (2015f). The coauthors contributed a review and discussions. In particular, they helped to implement key aspects of scenario SC 6 and SC 7 from Table 10.2, took care of bugfixing, and contributed test-cases for several of the scenarios from Table 10.2.

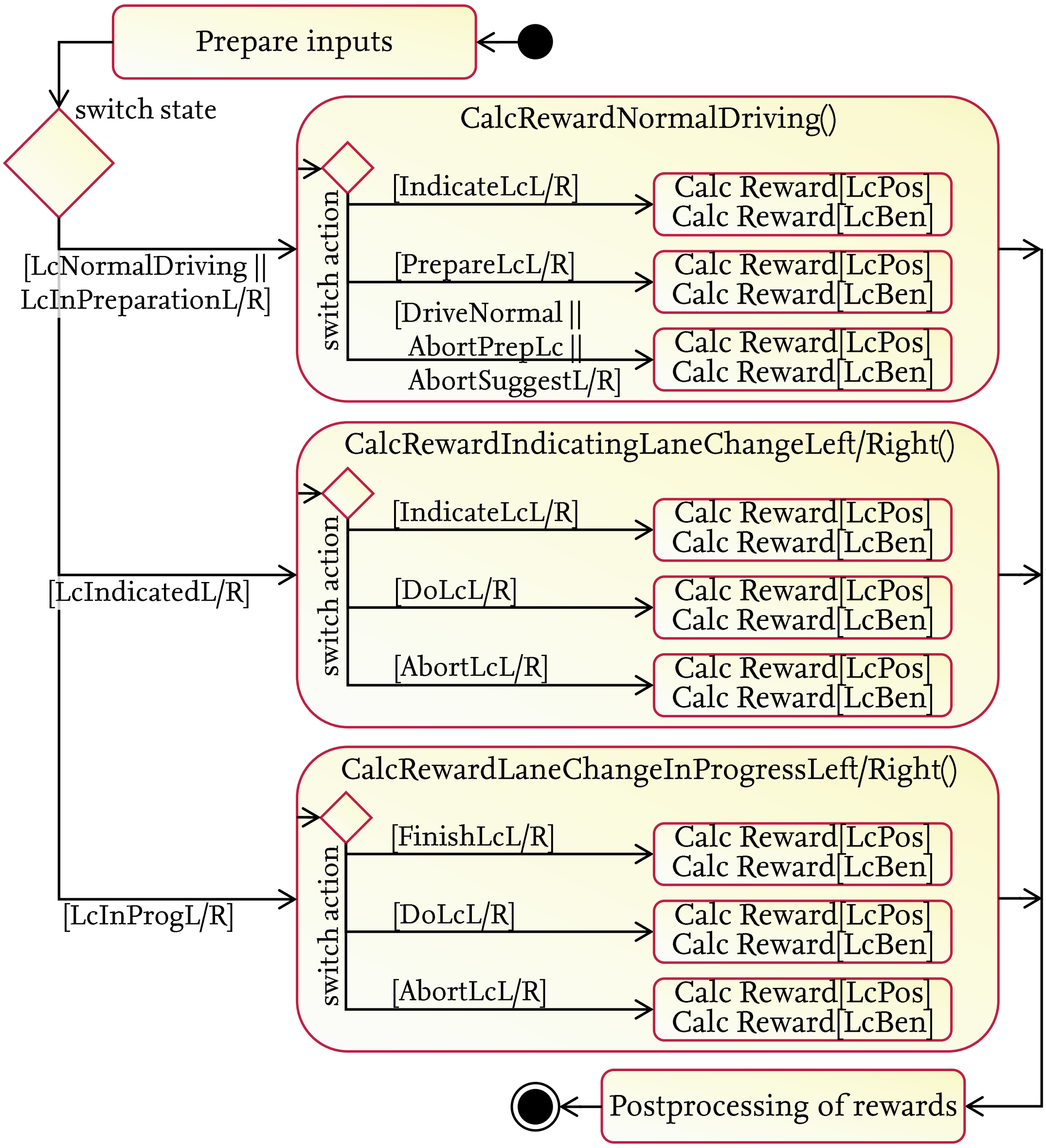


Figure 10.25: Simplified illustration of the reward function $\vec{r}(b(t_n), u(t_n))$. (L = left; R = right; calc = calculate)

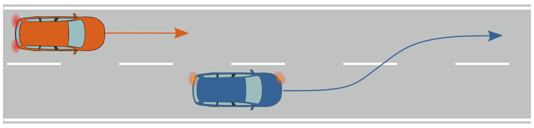
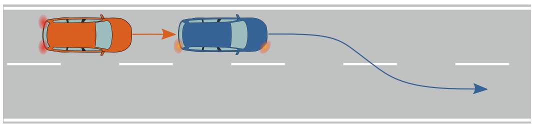
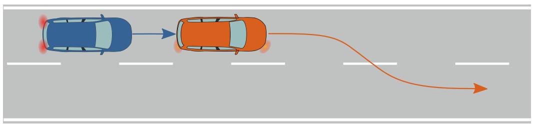
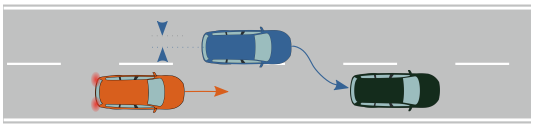
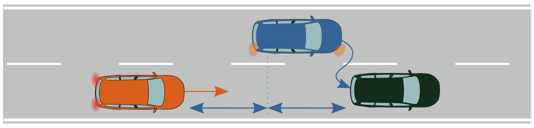
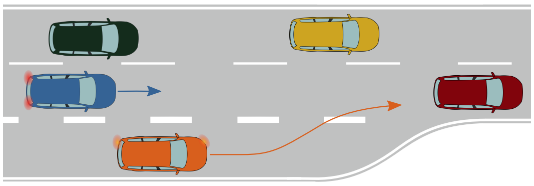
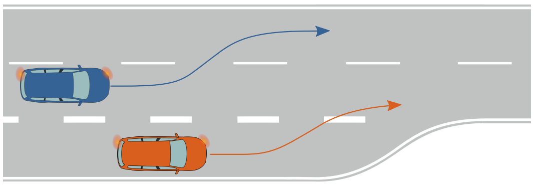
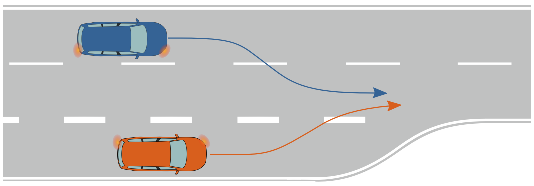
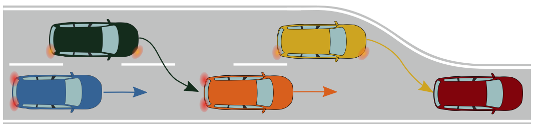
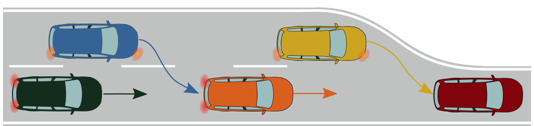
nication and awareness channels (cf. section 4.2) is communication by intended and unintended gestures. As pointed out in section 4, making use of these channels imposes extremely high perception requirements, which are currently not met. As a consequence, cooperation is limited to very few scenarios, for which communication and awareness channels exist in today's automated vehicles.

Chapter 4 raised the question of what scenarios can be solved by "cooperative" maneuvers with today's environment perception power and without omnipresent V2X equipment. Table 10.2 restates these scenarios and summarizes their consideration of the current state of the implementation.

Considering Discomfort for Rear Vehicles

Considering (dis-)comfort costs for rear vehicles is addressed through the consideration of the hidden state variable, modeling whether a lane change is possible due to the dynamic situation. The Bayesian network uses the necessary deceleration for

Table 10.2: Addressed cooperation scenarios for lane changes (ego vehicle in blue)

Scenario	**Illustration**	**Considered**
SC1: Considering (dis-)comfort costs for rear vehicles		✓
SC2: Giving way to pressing rear vehicles		✓
SC3: Tailgating slow front vehicle		✗
SC4: Squeezing into gaps by lateral offsets to the lane center		✓
SC5: Squeezing into gaps by longitudinal adjustment to gaps and usage of the indicator		✓
SC6: Letting vehicles merge in front if their lane or on-ramp ends soon		✓
SC7: Clearing a lane for vehicles if their lane or on-ramp ends soon		✓
SC8: Not changing to a lane on which another vehicle is about to merge to		✓
SC9: Dedicated handling of zipper method merging where the automated vehicle is letting merge		✓
SC10: Dedicated handling of zipper method merging where the automated vehicle merges		✗

a vehicle approaching the automated vehicle from behind. It changes the tolerable decision threshold based on the automated vehicle's state, whether a lane change has been indicated or initiated already.

Giving Way to Pressing Rear Vehicles

Although it is not legal (at least, e.g., in Germany or most US/Canadian jurisdictions), tailgating a slower vehicle is a common way to communicate one's intention to drive faster than the front vehicle. As an automated vehicle will have to stick to the legal traffic rules, it will – in the author's implementation – not speed and thus will be faced with a pressing rear vehicle quite often. Therefore, a pressing rear vehicle is detected based on its distance and time gap and the discomfort for the automated vehicle of clearing the lane to the cost of changing to a lane with a slower traffic flow velocity. If the discomfort for the automated vehicle is acceptable the automated vehicle will clear the lane for the pressing rear vehicle.[28]

Tailgating a Slow Front Vehicle

The opposite aspect of the aforementioned cooperation scenario, tailgating a slow front vehicle, is not implemented for two reasons: On the one hand it reduces the safety margin for reacting to sudden braking maneuvers, while on the other hand it is – at least in Germany – simply forbidden by law. The discomfort of the passengers to drive slower than the desired velocity is accepted in favor of maintaining a higher time gap to the front vehicle.

Lateral Offsets

To change lanes in very dense traffic it is quite common that there is simply no long enough gap on the neighbor lane for a lane change. It is necessary to use the cooperative driving behavior of other vehicles in order to merge to such a crowded lane. A common way to stipulate such cooperative behavior is to drive off-center towards the lane a vehicle intends to change to. If a vehicle on the neighbor lane is cooperative, it will open up a gap. If it is not willing to cooperate, it will not open or will even close a gap. Thus, such a cooperation request by driving behavior will make it easier to identify a good gap to merge to. Given a gap for merging can safely be identified, a lane change will be indicated and executed. [29]

Longitudinal Adjustments and Indicator Usage

If the need to change lanes is becoming very urgent, it is beneficial to activate the indicator even without a suitable gap for a lane change. This will stress the urgency of a lane change maneuver and might increase the willingness of other vehicles to cooperate. Vice versa, it renders the selection and approach to another gap than the direct neighbor gap less coherent. Thus, it reduces the number of behavior options the automated vehicle will have. Therefore, the urgency of the lane change maneuver is used to affect the behavior of the lane change decision making. As long as the urgency is low, the automated vehicle will evaluate the gap qualities as in section 10.3.3 and will try to align itself longitudinally towards the center of the best gap. If the urgency increases or if a gap alignment is only possible with extremely strong braking, the indicator will be activated.

[28] Details of the implementation are presented in the invention disclosure "Fahrstreifenwechsel bei Hinterfahrzeug mit Überholabsicht (English title: Lane changes with the consideration of rear vehicles with an intention of overtaking) (Ulbrich et al., 2015d)."

[29] Details of the implementation are presented in the disclosure of the invention "Anzeigen einer taktischen Fahrstreifenwechselabsicht durch Fahren mit lateralem Offset (English title: Indicating a tactical lane change intention by driving with a lateral offset) (Ulbrich et al., 2015b)."

A common scenario for driving on highways is to let traffic merge from on-ramps into the rightmost lane (cf. section 10.3.2). Particularly if big vehicles like trucks are merging on an on-ramp, they will assume some cooperation from other traffic participants and just merge. For an automated vehicle to handle such scenarios it is necessary to detect that the neighbor lane is a soon to end on-ramp, that there are dynamic objects on that ramp, and possibly even that these objects are trucks and have the indicator activated. With the current state of implementation it is not possible to obtain reliable estimates about the latter two information. For the first two it is necessary to have lane level localization information and a correct map. Reliably detecting the lane end a few hundred meters ahead visually was not possible with the state of the implementation. The aforementioned let-someone-merge lane change situation is a situation in which the automated vehicle has two options for cooperative behavior. Either it can decelerate to let someone merge or it can clear the lane with a lane change. The first behavior has been implemented in a longitudinal control module, which is not discussed in this thesis. The latter feature has been implemented for the lane change planning.

Letting Vehicles Merge

Clearing a Lane for Mergers

A slight variation of the previously mentioned scenario is a merging scenario, where a dynamic object intends to merge from an on-ramp to the rightmost lane of a highway. The automated vehicle is assumed to drive on the second rightmost lane. In order to prevent two vehicles, the one on the on-ramp and the automated vehicle from changing lanes to the rightmost lane of the highway, the automated vehicle is not allowed to perform a lane change to the rightmost lane in on-ramp areas of a highway. This cooperative behavior is implemented by ability restrictions as described in section 10.3.2.

Not Changing Lanes when others Merge

The last cooperative scenario presented in Table 10.2 is the dedicated handling of a zipper-method merging situation. It is used if the number of lanes is reduced. According to traffic laws (e.g., §7 Sec. 4 StVO, 2013), both lanes shall be used until the merging point and, here, vehicles from the two lanes shall proceed in alternating turns. In the current state of implementation there is no dedicated counting of vehicles to determine their merging order. There is some kind of indirect handling of this scenario by combining the features of the advanced adaptive cruise control to react to merging vehicles on the neighbor lane and the ability of the automated vehicle to squeeze into gaps by longitudinal adjustments and lateral offsets. However, at the moment the automated vehicle will try to pursue an early merge strategy rather than a late merge strategy. This helps to simplify the merging process and to increase the chances of the maneuver being executed successfully. Dedicated handling of such a cooperative scenario fits well with the concept of planning-ahead decision making. A situation prediction is perfectly suited to determine the order in which vehicles will merge. However, since the scenario has already been quite well resolved by the longitudinal control algorithms and the gap adjustment no additional effort has been made here.

Dedicated Handling of Zipper Method Merging

10.5.2 Additional Issues

Tactical behavior planning for lane changes entails several additional aspects that can only be covered on a very abstract level in this thesis.

Lane Change Aggressiveness

How aggressively versus how comfortably a lane change needs to be executed depends largely on the available maneuver space. If a lane change is executed in free-flowing traffic, it can be performed with little lateral acceleration. If it needs to be executed as part of traversing, e.g., a highway interchange or in order not to miss a highway exit, less maneuver space is available. Hence, lane change criticality and planned maneuver times are calculated and translated into scaling comfort parameters for the underlying trajectory planning algorithms.

Revising Decisions

Revising former decisions will always be perceived as inconsistent behavior. Yet it may be the single most important feature for behavior execution based on uncertain perception data. To ensure consistency, it is necessary to implement some hystereses in the reward model.[30] Both the lane change possible and the lane change beneficial estimates transformed by a non-linear transfer function are based on the "lane change status" (cf. section 10.2.1). The reward from just getting to the alternate side of such a hysteresis curve needs to be discounted over the planning horizon to ensure behavior consistency.

Indicator for Gap Adjustment

As mentioned in the previous section, flashing the indicator is used to cooperatively open up gaps after they have been identified as a targeted gap. Activating the indicator will increase the chance that a lane change is ultimately successful because other drivers may specifically open a targeted gap for the automated vehicle. Moreover, driving at low speed on a lane with faster traffic flow only to merge towards a slower lane without flashing an indicator will not be understood by other drivers and may result in being honked at or tailgated. On the other hand, passengers in an automated vehicle as well as other traffic participants will judge a driving behavior as inconsistent if the indicator is activated multiple times to target several different gaps. Hence, a gap adjustment for lane change preparation is a lot more subtle and easier to overthrow if it has not yet been communicated by indicator activation.

Deciding whether or not to activate the indicator during a gap adjustment depends on the current velocity deviation from the typical lane flow velocity and the jerk induced by braking/accelerating for that gap adjustment. If a lane change becomes more urgent in order to not miss the navigation goal, the gap adjustment is allowed to trade driving comfort against maneuverability. This will result in more jerk and thus indicator activation.

Predictive Indicator Activation

Apart from a gap adjustment, indicator activation can also be triggered by a-priori information from a map. Such predictive indicator activation greatly simplifies merging and highway interchange maneuvers. It is relevant because intentions can be communicated prior to actually reaching and perceiving, e.g., a neighbor lane in a weaving area. Given the look-ahead capability of today's lane camera systems and the time it takes to set up a stable track of a new lane at the side, a neighbor lane is often only detected when the automated vehicle has reached its beginning. Without prior flashing of the indicator, other vehicles may transition directly to such a neighbor lane and may require complicated and maneuver space consuming gap

[30]The need of some kind of hystereses is a necessity when dealing with noise in measurement data. Yet, it is likewise a sign of some insufficiency of the chosen model. The fewer hystereses are necessary and the later they are applied in the signal processing chain the smaller will be their side effects.

adjustments. Predictive indicator flashing makes transparent what the automated vehicle intends and helps other traffic participants to adapt accordingly.

Indicate Lane Change Abortions

Thirdly, the indicator may be activated in an opposite direction to communicate the abortion of a lane change maneuver. A lane change starts not at the point where the automated vehicle crosses a lane but rather when it starts to move away from the center of the initial lane. Despite sounding trivial, it is absolutely not trivial when to flash back and when not in case of a lane change abortion. This is because a lane change abortion is entirely transparent due to the fact that the automated vehicle flashes in the opposite direction. Thus, "hiding", e.g., lane perception issues during lane changes by the inertia of mass of the automated vehicle no longer works. Whether or not a lane change abortion is announced or executed secretly depends on the lateral displacement to the center of the initial lane, whether a lane change abortion is due to a fast rear vehicle on the neighbor lane or not, and how long a lane change has already been indicated in the initial direction.

Marginal Integral Velocity or Travel Time Gain

Specifically for driving on US highways with low relative velocity differences and typically sparse traffic, a marginal integral velocity (or travel time) gain feature has been implemented and tested.[31] As described in section 10.3.1, the dynamic environment is considered for estimating the benefit of performing a lane change. Yet, specifically for the US there is often a truck driving marginally slower than the speed limit, for example. Yet it is not slow enough to justify an immediate overtaking. For this, a long-term marginal integral velocity gain is calculated over a longer time horizon. This emulates a human driver being sufficiently annoyed by such a marginally slower front object.

10.6 Conclusions

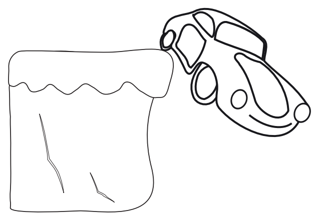
This chapter described the core ideas for the tactical lane change planning approach. The implementation has been structured into three main models and one behavior planning core: A measurement model to translate perceived information into a best possible state estimate, a situation prediction model for predicting a situation for a certain time in the future, and a reward and cost model to assign rewards for executing specific actions in specific situations.

What is missing?

Neither this chapter nor a dissertation as a whole can drill down to implementation specific details. The chapter clearly illustrates that there are numerous special cases to be considered for coming at least close to human driving behavior. For an implementation, it would be nice to aggregate all these behaviors into a rule base to be evaluated as in a logical solver. Yet, at the time of writing, it seems overly ambitious to obtain a real-time capable, uncertainty-tolerant solver-implementation that is performance-wise close to the solution presented here.

[31] Details of the implementation are presented in the disclosure of the invention "Marginale, integrale Vorteilhaftigkeit bei der Bewertung des Überholnutzens eines geringfügig langsameren Vorderfahrzeugs (English title: Marginal, Integral Benefit for Evaluating the Overtaking Utility of a Marginally Slower Front Vehicle) (Ulbrich et al., 2015c)."

11 Critical Discussion of Limitations [1]



This part has presented the approach for tactical lane change behavior planning developed in this thesis. The approach has successfully been implemented and demonstrated in public traffic. Yet, it does by no means *solve* the issue of tactical lane change behavior planning.

The core issue of this thesis is to plan tactical lane change behavior under uncertainty. According to section 7, the requirements are rapidity, consistency, providentness, determinism, and complying with values. From a conceptual point of view there are several limitations.

Context Modeling

At the beginning of this part the concept for lane change context modeling has been presented. Section 8 highlighted three issues. Firstly, that concept of a snapshot is to a certain extend voided by derivatives to describe dynamic elements' trajectories. Secondly, that the separation of a scene or situation prediction is currently not fully elaborated. Thirdly, that the scene description may not yet be complete. For the author there is no denying in these limitations.

Value-Orientation

Yet, the limitations regarding value-oriented behavior planning are far more severe and fundamental. Chapter 9 highlighted the challenge of the consideration of values on the one hand and bridging the gap towards a code implementation on the other hand. This highlights the core essence in this thesis: Behavior planning under uncertainty. This entails perception, execution, and prediction uncertainty. All three have an influence on behavior decisions and values. In a sound ethical framework decisions are free of uncertainty. In the real world uncertainty is there.

Conceptual Limitations

Section 10.1 related the here presented implementation to existing concepts from the model predictive control and the partially observable Markov decision process research community. In this thesis, it is neither assumed to have fully value-discrete state spaces and fully value-discrete observations nor to have fully value-continuous state spaces and fully value-continuous observations linked by linear models. This is a severe limitation because that is why much existing research from the POMDP and MPC research community cannot be leveraged on.

A second limitation resulting from not using value-discrete state spaces and observations is, that it is harder to use machine learning approaches to learn for instance the measurement, situation prediction, or reward model (cf. section 10.2.2).

The here developed concept of tactical behavior planning is related to the DESPOT approach for online decision making. By using parametric probability description instead of a Monte Carlo sampling approach the requirement of computational efficiency (cf. rapidity) is met. Yet, a limitation is that generality is lost. Particles as in Monte Carlo sampling could be used to fit *any* probability distribution. The

[1] Special thanks to Prof. Markus Maurer for helping to identify several of the here listed limitations.

here chosen parametric probability distributions are a limitation if a dimension of the belief space is not well approximated by a normal or beta distribution.

Several application specific heuristics are used to simplify the overall planning problem. Hence, the here developed approach addresses the issue of lane change planning but loses generality for other application domains.

Measurement Model

This part described how the overall problem is broken down into a measurement model, a situation prediction model, and a reward model. The measurement model translates observations into belief estimates. The overall problem is separated into a lane change beneficial estimation, a lane change possible estimation, and a gap quality assessment. The lane change beneficial estimation is structured along the "desires" for a lane change due to the dynamic situation, due to the infrastructure situation, and due to timing restrictions (cf. section 10.3.1). Yet the approach does not relate those towards more aggregated motives for behavior.

Lane Change Beneficial

Within the lane change beneficial evaluation, the evaluation of the dynamic situation makes use of several heuristics (selection of relevant objects, filtering, switching between modes and blanking out certain aspects, ...). These proved to work well, but they lack a sound theoretical foundation. The evaluation of the infrastructure for a lane change benefit ignores several aspects that have been discussed in the literature already. This entails handling HOV lanes, the lack of learning individual driver preferences, or the avoidance of cooperation scenarios (cf. section 10.3.1). Last of all, the concept of timing restrictions is used to achieve a human driver like behavior (cf. section 10.3.1). For many of the here addressed scenarios, a human driver would not use a timing restriction but would rather resolve those with situation specific reasoning. The necessity of these timing restrictions may be considered as a result of the lack of such a situation specific reasoning. So far, rules are implemented to derive beliefs about states, intentions, etc. Yet, there is no truly *intelligent reasoning*.

Lane Change Possible

The lane change possible evaluation is based on a set of analytic equations to calculate necessary decelerations with simple case distinctions based on relative velocities. Accelerations or jerks are ignored entirely. This is a significant simplification of human situation assessment. The lane change possible situation assessment uses measurement uncertainties as part of the perception uncertainties for a rather conservative situation assessment, but ignores existence uncertainties or association uncertainties. Likewise, the skill and ability monitoring estimates if general limitations currently or permanently apply to behavior decision making. Yet, this aspect is still in its infancy for lane change behavior planning. Currently, selected aspects like a degraded sensor viewing range would be considered but no general degradations of the system. Here future research is needed.

Gap Quality Assessment

Even if a lane change is currently not possible it is necessary to evaluate where a best possible gap is. This information is generated by gap quality assessment and used for a subsequent adjustment towards such a gap. Simplifying heuristics are used for gap quality assessment (cf. section 10.3.3). The gap quality assessment is executed for ten gaps around the automated vehicle. The more sophisticated lane change possible estimation only immediately around the ego vehicle to the left and right. Here the heuristics could contradict each other causing a gap adjustment to a gap that will then be insufficient for a lane change. Likewise, temporal inconsistencies in the gap assessment may cause inconsistent swinging between gaps.

Uncertainties

Section 10.3.4 addresses how *measurement uncertainties* as part of the *perception uncertainties* are translated into uncertainty of belief distributions. This addresses *measurement uncertainties* but currently ignores *existence uncertainties* or *association uncertainties*. So far, an existence uncertainty is derived from the track age (cf. section 10.3.5) and used in the lane change beneficial estimation. In particular the incorporation of a more profoundly estimated existence uncertainty is assumed to provide significant performance improvements towards the lane change situation assessment.

Situation Prediction

The situation prediction model presented in section 10.4 estimates how a situation evolves into the future. The better the prediction is, the better will be the hereof resulting behavior decisions. A commonly used situation prediction model from the literature has been applied. Yet, preempting the evaluation in section 15.3 the situation prediction model does not seem far superior to far simpler prediction models. To the author this is an indicator that the situation prediction model has even conceptual insufficiencies. It is a model that provides a theoretically sound prediction of correct state estimates. Yet, it seems to not appropriately allocate resources towards the root causes of prediction errors. While measurement uncertainties are propagated through the prediction model into the future there is no explicit consideration of existence uncertainties or association uncertainties. Yet, all these aspects of the perception uncertainty translate into prediction errors. Last of all, the situation prediction model currently cannot estimate prediction uncertainties resulting from the "inherently unpredictable future of the environment" as introduced in section 2.5. Even if intentions were know and predicted correctly, disturbances to the traffic system would render predictions to be inherently wrong. Section 2.5 introduced the example of a moose running on a street to cause a prediction uncertainty. Such prediction uncertainty is currently not estimated.

Reward and Cost Model

The reward model and cost model (cf. section 10.5) is used to evaluate an action in a particular (future) situation. A severe limitation of the reward model is that it cannot analytically be written as a set of equations. It addresses several special situations in which it is necessary to use particular aspects of the situation assessment in a certain way. As a result of this it cannot simply put into a sum of a few summands. Moreover, it is highly application specific. Relevant factors, weights, and case distinctions were found by manual tuning. To the author, this is a prime example for the application of offline machine learning to find a better reward model.

Execution Uncertainty

At last, a limitation is that the execution uncertainty is only marginally covered. According to section 2.5, it entails uncertainty from wear-and-tear, control noise, and mechanical failure as well as errors in internal models and algorithmic approximations. Certain aspects are covered implicitly. For example section 10.3.3 considers the likeliness to reach a certain gap. Likewise, the skill and ability monitoring entails a certain amount of system monitoring regarding the failure of relevant components. Yet, currently the skill and ability monitoring does not cover the failure or degradation of mechanical components.

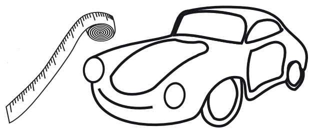
Insufficiency for SAE Level Five

All in all, insufficient handling of perception, prediction, and execution uncertainty introduces a severe limitation. For a research project, this simplification is often implicitly done and accepted. Due to limited resources it is considered sufficient that the system works at least *most of the time*. Rare events like a moose running on the street, or a broken steering system are currently left to the responsibility of a safety driver. For performing show case demonstrations this is sufficient. Yet, these are exactly the issues that need to be addressed for implementing lane change planning in an automated vehicle handed over to customers. For the sake of scientific honesty and to countervail the media impression that only legal reasons are among the last challenges towards automated driving, those issues are mentioned here. For a SAE level three to five system (cf. section 2.1) this will not be acceptable and opens a wide field of future research.

Part III

Metrics and Evaluations

12 Methods and Metrics for Performance Evaluation



This chapter presents methods and metrics to quantify the performance of lane change planning algorithms. After a short overview over possible methods and metrics to quantify an overall behavior, the focus is put on metrics for behavior planning in automated driving. To conclude this chapter, those methods and metrics are linked to the requirements of Chapter 7.

12.1 Methods and Metrics to Measure the Overall Behavior

Performance Evaluation in Psychology

The issue of performance evaluation is not unique to the field of automated driving. In fact, it is similar to the evaluation of a human agent's performance. In the field of psychology, situation awareness as a key part of the overall behavior is primarily measured by (a) task performance, (b) memory probes, or (c) (subjective) self-assessment and assessment by others (Seitz, 2015).

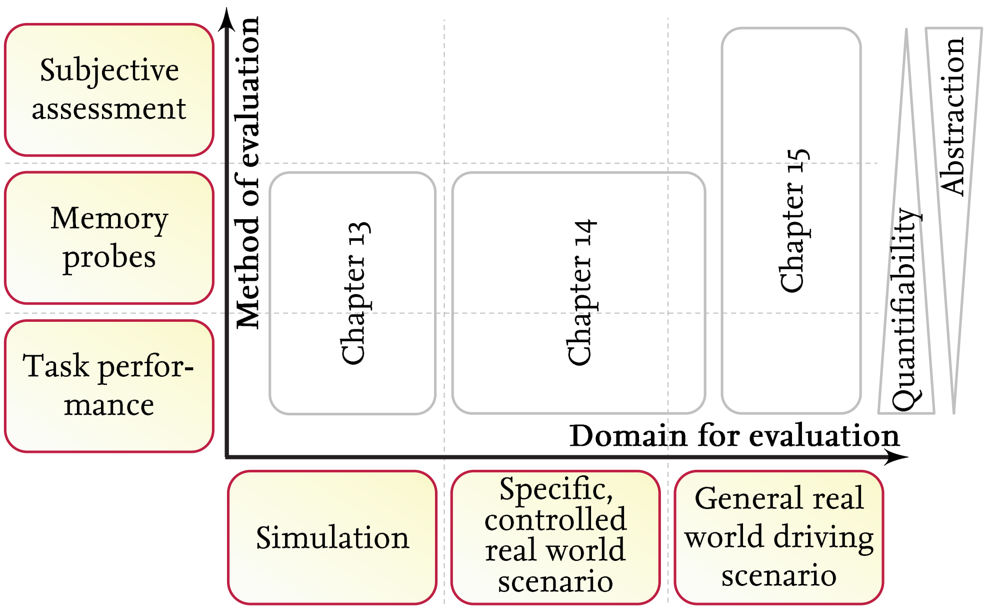


Figure 12.1: Matrix of evaluation methods and domains

Task Performance

Task performance is measured by a demonstrated behavior in a – to be accomplished – task and quantified by metrics such as reaction time, number of errors, number of control movements, etc. The equivalent of this in automated driving would be to expose an automated vehicle to controlled scenarios and to quantify its performance with scenario-specific metrics.

Memory Probes

Memory probes are questionnaires which are to be filled out during a – to be accomplished – task at specific points in time. The experiment is interrupted and questions are asked about past details of the – to be accomplished – task. While questionnaires do not seem eligible to be answered by a technical system, the underlying idea of accessing the internal state estimates of the system also seems applicable.

Subjective Assessment

In psychology, (subjective) self-assessments and assessments by others are conducted by questionnaires during or after an experiment. A self-assessment for a technical system requires the ability to reflect upon its own behavior and is thus limited by the cognitive capabilities of the system itself. Vice versa, an assessment by others is achievable, yet requires significant resources in terms of time and money.

12.2 Review of Metrics for Behavior Planning Performance Evaluation in Automated Driving

In the automated and assisted driving community, to date relatively little research has been conducted on metrics to measure overall behavior performance. So far, many metrics have focused on the aspect of risks.

Risk-Based Metrics

Every driving maneuver and driving decision inevitably results in some risk, either regarding its safety or its complete executability. Eidehall & Petersson (2008) present a statistical framework to assess traffic situations with two other traffic participants for longitudinal collision warnings. They assess a set of control actions regarding the risk level of a collision with one of the other vehicles. They weight the risk of a collision with other vehicles by their visibility. Based on this, they define a warning thread level α. To quantify the collision risk of an action numerically, they use Monte Carlo Sampling. An evaluation has been done in some simulated scenes and, ex post, on a set of recorded measurement data. The data has been labeled manually and the false positive and the true positive rates are presented.

Maximally Available Reaction Time

Winner et al. (2013) provide a more general framework to come up with specific risk metrics to quantify the safety gain from driver assistance systems. In this publication, the authors mainly focus on longitudinal collision warnings. However, they do discuss how to transfer those metrics to a lane change situation assessment. They stress the advantages of using Habenicht's (2012) maximally available reaction time $\tau_{react,max}(t)$ over using simpler metrics like the time gap or the time-to-collision.

Kopf's Time Reserve

To quantify hazards, Kopf (1993, p. 48 ff.) uses the inverse of the time reserve until a hazard-mitigating action has to be initiated to avoid a collision. He identifies a situation aspect and an action aspect of assumed hazard-mitigating action(s) and formulates "hazard functions" for these. Kopf suggests to calculate the integral of a hazard function over an evaluation time interval as an numeric measure of absolute hazard.

Ardelt et al. (2012) provide a macroscopic evaluation of their implementation in a real, automated vehicle over 65 km on a public highway. In total, 32 lane changes were executed and no maneuver was aborted. In total, the system is said to be tested for "several thousand kilometers" (Ardelt et al., 2012, p. 1584). Ardelt et al. (2012, p. 1584) mention that "the system waits for a considerably large traffic gap to not obstruct other traffic participants during an overtaking maneuver." Here, it is open to provide metrics on how "human like" the lane change behavior is. In the extreme, waiting for gaps that are, for instance, 100 m long may allow to never abort a maneuver, but on the other hand, it does not help at all to make a passenger pleased in dense traffic. Apart from a macroscopic perspective, Ardelt & Waldmann (2011) evaluate their algorithms by presenting system state variables during the execution of single lane change maneuvers.

Macroscopic Evaluation in Traffic

Quantitative and repetitious metrics exist for evaluating simulation models: Several driving and lane change behavior simulation models have been developed and evaluated (cf. section 6.2.2). Their applicability is often shown by "reality like" macroscopic behavior in a simulation environment utilizing such a model. Here, macroscopic behavior refers not to individual maneuvers but rather to aggregated behavior over a long stretch of driving. Gipps (1986, p. 411) conducts such a macroscopic analysis of 2000 vehicles over the course of 8 minutes on a three lane road. It illustrates the feasibility of the lane change model by demonstrating the robustness of the system against disturbances. Kita (1999, p. 311) validates his game theoretic lane change behavior model with video-taping and by labeling the lane change behavior in a real world on-ramp merging situation. Hidas (2002) uses macroscopic data of real world average speeds in highway merging situations as a function of the total traffic flow to compare his simulation model with real world behavior. Hidas (2005) refines this analysis for merging situations by evaluating the gap size as a function of a merging vehicle's speed difference, the average rates of vehicle stops as a function of the traffic flow and the average speed as a function of the traffic flow for weaving situations at highway interchanges. Shen & Jin (2012, p. 273 ff.) compare three lane change simulation models back-to-back in their simulation environment. They evaluate the models' computational complexity by comparing the simulation time per frame as a function of the number of simulated vehicles. Moreover, they compare the number of occurring "free" (dynamically beneficial) and "imperative" lane changes (directly induced by the mission goals) as a function of the traffic density. Finally, they perform a scenario-specific microscopic analysis of the course of a lane change by comparing vehicle velocities, gap sizes, and lateral offsets during the execution of a lane change.

Macroscopic Behavior Metrics in Simulations

Microscopic State Variables

Siedersberger (2003, p. 149 ff.) and Pellkofer (2003, p. 125 ff.) evaluate their implementation for automated driving by providing real world measurements of state variables for selected maneuvers such as lane-keeping, following a leading vehicle, stopping, or turning. Furthermore, they illustrate the execution of an abstract mission by providing a time series of system state variables over the course of that mission. They did not develop aggregating metrics to quantify the performance of the tactical behavior planning.

Microscopic State Variables within a Maneuver

Specifically for lane changes, Naranjo et al. (2008) chose a similar approach for their evaluation by presenting the microscopic process of an overtaking maneuver by

Microscopic State Variables for Overtaking

illustrating the specific state variables of two vehicles involved in the maneuver. They illustrated the vehicles' speed, time gap, and lateral displacement as a function of the longitudinal position since the start of the maneuver.

Time Gap and Time-to-Collision

In the field of driver assistance systems for lane change assistance, Chen (2009, p. 109 ff.) focuses on an evaluation with a pool of test persons and uses an assessment based on time gaps and time-to-collisions. Mammar et al. (2006) provide a review of the usage of time-to-line-crossings for lateral risk assessment in driver assistance systems. Habenicht (2012) covered a similar application as in Chen (2009) in his thesis. He developed a Human-Machine-Interface (HMI) for such lane change applications and evaluated a developed lane change assistance system regarding driver stress and safety assessment. Freyer (2008) uses a group of test persons for a subjective assessment of an adaptive cruise control being improved for lane change situations and a comparative analysis of scenario-specific task performance metrics. As metrics, he uses sheer-out distance, sheer-out time gap, sheer-out difference velocity, lane change duration, lane choices, and the number of braking actions with more than $-1.5\,\mathrm{m/s^2}$.

Ambiguity

Similarly, Schubert & Wanielik (2011) evaluate an implementation for an overtaking maneuver by illustrating the microscopic state variables of two vehicles within the overtaking process such as longitudinal and lateral positions, velocities, and expected utilities for different actions during the course of the maneuver. The evaluation is performed on both perfect simulated data and real world data from driving in traffic. In Schubert (2011, p. 136) and Schubert (2012) the author also evaluates the ambiguity of the obtained decision. He defines this as a metric for the information entropy over the decision alternatives. If the expected utility of one decision alternative dominates all the others, the ambiguity is $H = 0$. If all alternatives have a similar expected utility, the ambiguity is $H = 1$. To develop the system behavior, he uses deceleration-to-safety time (DST) as a central metric (Schubert, 2011, p. 129).

Deceleration to Safety Time

Reichel (2013, p. 161 ff.) uses labeled, recorded measurement data to train and evaluate an algorithm for situation assessment in merging scenarios. Frese (2012) evaluates cooperative driving maneuvers in a simulation environment. He chooses the computation time and the percentage of correctly resolved situation plans as performance metrics.

UCSD

Sivaraman & Trivedi (2014) evaluate their driver assistance system for recommending lane changes and accelerations/decelerations for lane change preparation in 50 merging maneuvers and 100 lane changes on highways and multi-lane urban roads. They characterize a scenario by the ego vehicle's dynamic state regarding speed and acceleration and evaluate the merge planning by an aggregated number of acceleration/deceleration recommendations. They do not quantify consistency but provide a detailed understanding by showing time series data for single maneuvers.

It would be best if it were possible to evaluate behavior planning as good as a computer vision algorithm can be evaluated by standardized performance metrics and reference data as in the KITTI dataset (Geiger et al., 2012). Currently, it is lacking adequate ground truth label data to allow an evaluation of behavior planning.

12.3 Metrics in this Thesis

In Chapter 7, the author listed rapidity, consistency, providentness, determinism, and compliance with values as meta-level requirements for tactical behavior planning. How could fulfilling these requirements be quantified by metrics?

Rapidity relates to computational complexities and induced latencies for new measurement data. Section 15.1 evaluates both by considering CPU loads and execution cycle times. Given that the implementation has plenty of buffers regarding computation resource restrictions, no thorough evaluation was necessary. Rapidity

Consistency can be quantified on two levels: Overall behavior consistency and the consistency of a situation assessment with a ground truth reference. Consistency

Overall behavior consistency simply counts the total number of lane change abortions versus successful lane change executions as a metric. Such an overall behavior consistency evaluation on a macroscopic level is provided in section 15.2. Under ideal conditions, such an evaluation should be reproducible by exposing a system under test to similar, reproducible scenarios and traffic constellations. Yet, creating reproducible scenarios with the same level of complexity as in real traffic in a controlled environment is, at the time of writing, infeasible. Thus, in attempt to minimize biases by individual traffic participants and their behavior a huge number of situations obtained from real world measurement data has been evaluated.

Assessing consistency against a ground truth has been demonstrated by Ulbrich & Maurer (2014) for lane changes in urban environments. Performing the same assessment on highways is not trivial (cf. section 15.6) because there is often no single ground truth reference for best behavior. Obtaining ground truth situation assessments from different individuals often results in contradictory voting.

Assessing providentness directly translates to a performance evaluation of a situation prediction. The more accurately a situation is predicted, the more provident the accordingly planned behavior will be. It is far easier to quantify the accuracy of a situation prediction than that of actual behavior planning because the significance of the deviating behavior may depend on the situation itself. As a metric for situation prediction quality, a distance metric which considers deviations in (dynamic) element existence and its state variables is used. Details on the metric and the evaluation are presented in section 15.3. Providentness

Additionally, lane change behavior planning should be deterministic. No learning is involved in behavior planning. The decision behavior is fully dependent on a single system state vector (cf. section 10.2.1). There is no black box model; every behavior decision can be fully linked to specific situation aspects. Thus, the behavior planning is deterministic and should not be an insurmountable challenge for an ISO 26262 development process. Determinism

Last of all, lane change behavior planning shall comply with value dimensions such as safety, mobility, legality, or user and third party satisfaction. In fact, while there are metrics to quantify safety (cf. sections 12.2 and 15.4.3), it is hard to come up with metrics for other value dimensions. Mobility entails abstract aspects like Compliance with Values availability of the automated driving system in different domains and situations. Moreover, it entails mission fulfillment. Certain missions may not be executable if the automated vehicle cannot pass a highway interchange (cf. section 14.3). If a mission can be fulfilled, metrics like the total travel time will be of relevance. A travel time reduction could be quantified by comparing the total travel time of a system with and without the capability of performing automated lane changes. Yet, in real traffic such a metric would be susceptible to several external factors like congestion, temporal road works, etc. Hence, section 15.5 uses the difference of the ego velocity and the neighbor lane velocity to quantify a lane change velocity gain as part of mobility. The algorithms in this thesis do not intentionally violate laws. Thus, legality will not be evaluated. User satisfaction and third party satisfaction is hard to quantify. At least certain aspects of the user satisfaction are evaluated in section 15.6. All in all, the author of this thesis tries to come up with first metrics to quantify value dimensions in automated driving. Yet, this is merely a first step. To the author it is a field where significant future research is needed.

Comprehensive List of Requirements

Attachment A provides a more comprehensive list of requirements than the meta requirements discussed so far in Chapter 7. These requirements are grouped by functional requirements (cf. Table A.1), user interface requirements (cf. Table A.2), usability requirements (cf. Table A.3), and performance requirements (cf. Table A.4). To provide a qualitative assessment of what is currently possible, the list entails check marks or x marks for each requirement. Some requirements are marked with both a check mark and an x mark to indicate that this requirement has only been partially met. Currently not met requirements are commented in attachment A.5.

12.4 Conclusions

Summary

This chapter introduced evaluation methods and metrics for behavior planning. General evaluation methods for behavior planning have been related to approaches from psychology. A review of lane change specific metrics revealed several weaknesses of those metrics and evaluation methods. Hence, for each meta-level requirement of rapidity, consistency, providentness, determinism, and compliance with values the chosen evaluation method has been outlined.

Challenges and Limitations

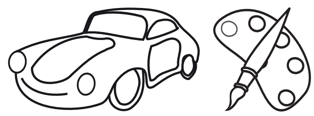
The challenges and limitations regarding metrics and evaluation methods for tactical behavior planning are numerous: First of all, an evaluation metric only provides its value of comparability if it is used by several implementations. So far, no one else uses all the metrics used here. Thus, a full comparison seems to be difficult.

Many groups have evaluated their implementations by illustrating state variables during single maneuver executions. Such an evaluation is likewise presented in this thesis in Chapters 13 and 14. This helps to understand how an implementation actually works, but barely provides any clear performance metrics that could help to compare different implementations.

A simulation-based performance evaluation in Chapter 13 provides reproducibility but, at the time of writing, cannot really quantify the performance criteria a human passenger easily "feels" in the car in real traffic. Vice versa, a microscopic maneuver evaluation in real world traffic imposes the challenge of maneuver reproducibility.

Yet, to the author, it still provides more value than the simulation-based performance evaluation. Last of all, a macroscopic evaluation remedies the dependence on single maneuvers by aggregating behavior over several hundred maneuvers. Ultimately, it is still biased by the experiment setup. Time of day affects traffic densities and the type of roads (two lane or three lane highway, close to metropolitan area or not, etc.) affects the overall level of difficulty and thus behavior performance.

13 Simulation-Based Performance Evaluation [1]



This section provides an overview of testing and simulation efforts to ensure the correct functionality of an item under test. The testing and validation efforts follow a four-step procedure as illustrated on the right in Figure 13.1. This four-step procedure can be integrated into the testing branch of the V-model[2] according to the checkmarks in Figure 13.1.

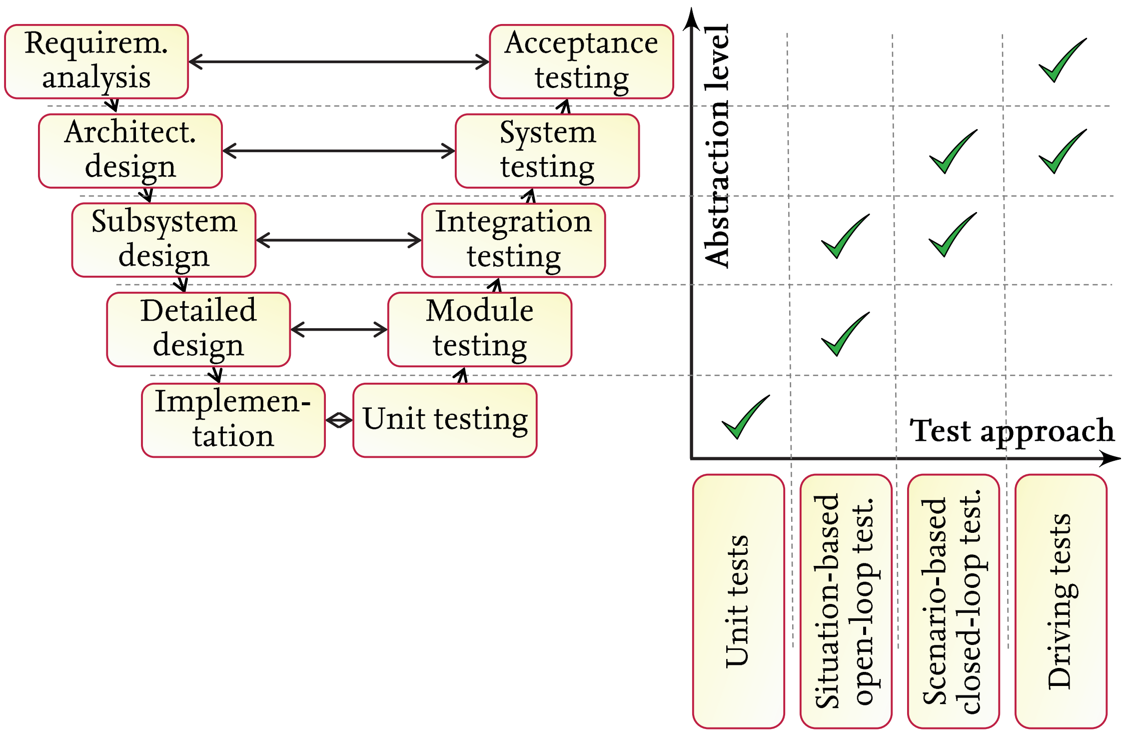


Figure 13.1: Four-step test procedure for software validation and testing (architect. = architectural, test. = testing, requirem. = requirement)

On a very basic level, unit tests are executed to test and ensure the correct functionality of aspects of atomic functions. As a next step, a situation can be generated as a stimulus to test a driving function as a whole in an open-loop test by situation-based testing. Such testing is limited in its scope to the driving function itself; it

[1]This chapter has been pre-published by the author in Ulbrich et al. (2017c). The coauthors contributed the implementation of several test-cases for situation-based open-loop testing. Further, they took care of providing a working simulation environment. Coauthors contributed the aspect of a test suite and test-cases to Figures 13.3 and 13.4. Moreover, they provided an in-depth review, helped to structure and review the article, and contributed in several discussions. Last of all, they took care of handling the article submission process.

[2]Cf. ISO (2011, Part 6), as depicted in Ammann & Offutt (2008, p. 6).

does not test the situation extraction from a scene. If this situation extraction also needs to be tested, the author proposes scene-based open-loop testing. However, in the test process for the lane change behavior planning for the *Audi A7 piloted driving concept* vehicle it was not necessary to implement the intermediate step of scene-based open-loop testing, because it is entailed in the later presented scenario-based closed-loop testing.

Levels of Testing

When all situation-based test-cases are passed successfully, scenario-based closed-loop testing is used to test an item under test in its interaction with strategic level modules and stabilization level modules as a whole. As a last step, testing is completed by real world driving tests. The test steps will be explained in detail in the following subchapters. The situation-based open-loop testing and scenario-based closed-loop testing is presented in the remainder of this chapter to test the lane change behavior planning module.

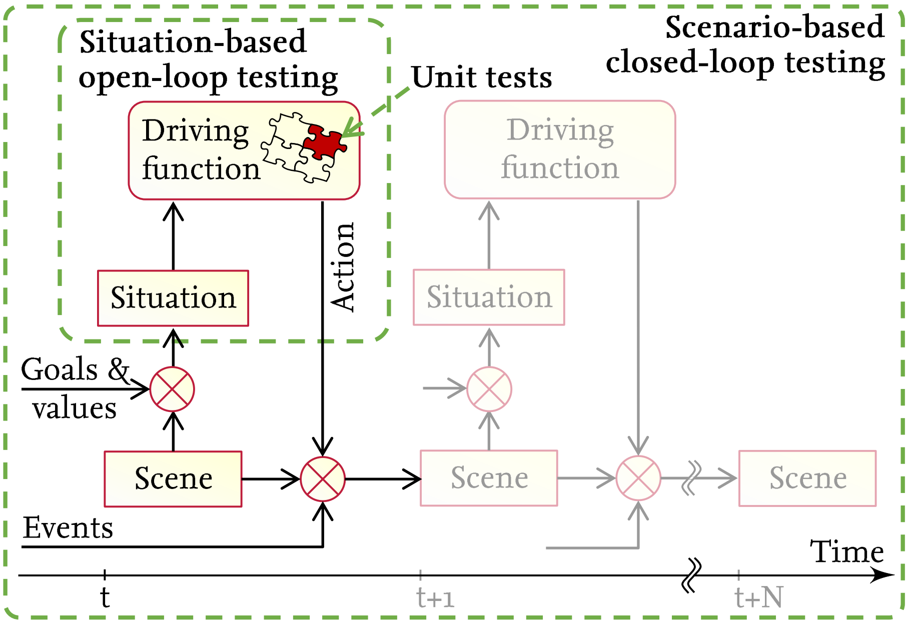


Figure 13.2: Illustration of the differences between unit tests, situation-based open-loop testing, and scenario-based closed-loop testing. The horizontal axis illustrates the scenario evolution over time

Figure 13.2 illustrates the differences between miscellaneous levels of testing. Unit tests only allow particular code parts of the driving function to be tested. This is depicted by single parts of a jigsaw puzzle of the driving function in Figure 13.2. Situation-based open-loop testing generates situations for one or several different timestamps from the test-case description and evaluates the behavior response without feeding this behavior response back into future situations. Scenario-based closed-loop testing specifies an entire scenario in a test-case. This includes scenes, events to alter the following scenes, and goals and values for situation extraction and as input for the driving function. The control and behavior response from the driving function is used to influence future scenes and by this implicitly future situations too. A test system generates the inputs in Figure 13.2.

13.1 Unit Tests

Unit Tests

On a basic level, unit tests are executed to test basic software functionalities. These unit tests are of particular value to reduce the number of errors in basic functions, e.g., to calculate distances, time gaps, time-to-collisions, or similar physical-law-based numbers that follow clear calculation rules and numeric properties. However, unit tests are not an eligible method for testing situation assessment functions as they typically require knowledge about the past development of a situation. Thus, it is difficult to evaluate the results of a single processing cycle of situation assessment. To test these more abstract functions, it is far easier to broaden the scope and to use situation-based testing instead of a traditional unit testing scheme.

13.2 Situation-Based Open-Loop Testing

Situation-Based Testing

Situation-based open-loop testing uses a broader scope for testing. While unit tests focus on testing single functions and lines of code, the focus for the situation-based testing is wider. According to Figure 13.3, a situation data structure is generated as a mock-up for a particular simplified real driving situation. The situation is fed unchanged into the tactical behavior planning module in the guidance block.

In the guidance block, the tactical behavior planning module evaluates the situation and derives tactical driving decisions accordingly. The driving decisions from the tactical behavior planning modules are compared with an a-priori-known ground truth of correct driving decisions. Deviations from that ground truth of expected behavior are evaluated and marked as a pass or fail of such a particular test. Other than in traditional unit tests, the same situation may be used repeatedly as a stimulus for a planning module. Thereby, steady states of dynamic, model-free filtering components (e.g., low pass filters in a dynamic Bayesian network) can be achieved and tested. Moreover, modules can be tested as a whole, not just single classes of them.

Advantages

An advantage of this test method is its applicability for fast, iterative testing of gradual software changes. Additionally, the test suite can easily be expanded with new test situations. Moreover, the tests can be executed faster than in real-time. Thus, situation-based open-loop testing is a versatile tool for testing during the development process and a necessary step to be executed and passed for any releases.

Limitations

A limitation of this test procedure is that the temporal development of a situation is currently not predicted by a situation prediction model. This renders it potentially insufficient for testing components containing model-based filters. Moreover, the approach is clearly limited by its open-loop nature: The situation is not modified and predicted based on the tactical behavior decisions of the module under test.

Table 13.1 illustrates three exemplary test-cases out of a test suite of 29 test-cases that are used for situation-based testing. Each of these situations is generated by a set of support functions. Each situation is fed into the tactical behavior planning module for planning lane changes for several cycles until a steady state of any low pass filtering component can be assumed. For the moment, each situation is repeated 400 times, resulting in an evaluation speed five times faster than real-time (3.2 s

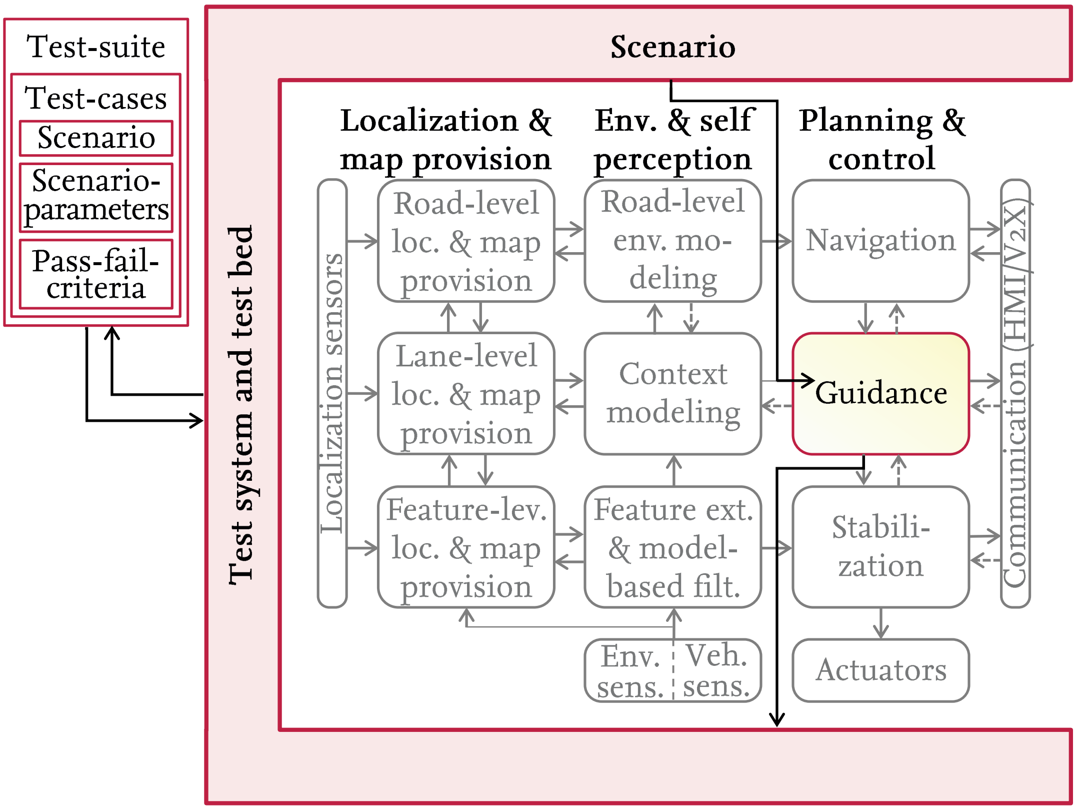


Figure 13.3: Situation-based open-loop testing as a middle ground between unit tests and scenario-based testing (env. = environment, ext. = extraction, filt. = filtering, lev. = level, loc. = localization, sens. = sensors, veh. = vehicle)

per test-case). After such a cycle the tactical behavior of the lane change planning module is compared to the expected behavior noted in the test-case. If the behavior is identical, a test-case is passed. The situation stimulus, introduced in section 8.2, contains value-continuous and value-discrete elements. To evaluate the behavior response of the lane change planning module, value-discrete behavior choices are evaluated.

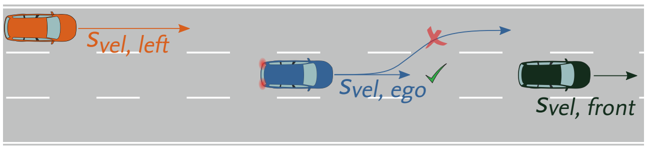
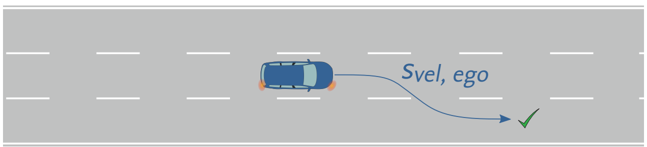
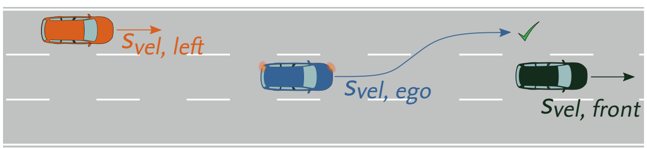
It is possible to execute a human-designed set of the 29 most essential test-cases on a standard computer in less than two minutes. This fact renders this test procedure very efficient for iterative testing, even after minimal source code changes. Apart from the test-cases themselves, Table 13.1 illustrates the expected resulting behavior for each of these test-cases and whether it matches the obtained resulting behavior from the lane change planning module. As indicated in the table, all three test-cases have been passed successfully.

13.3 Scenario-Based Closed-Loop Testing

Scenario-Based Testing

Scenario-based closed-loop testing lifts some of the above described limitations of the situation-based open-loop testing. A test-case in scenario-based closed-loop testing specifies an entire scenario with its parameters and pass-fail criteria. This includes scenes, events to alter these scenes, and goals and values used for situation extraction and as input for the driving function. The device under test is not only

Table 13.1: Three selected test-cases for situation-based lane change testing

Nr.	Test-Case (Initial Situation)	Expected Result	Passed
TC 22)	 The ego vehicle (blue) drives at $s_{vel,\,ego} = 33.3\,\mathrm{m/s}$ on the center lane of a three lane highway with no speed limit. An object (green) drives with a longitudinal offset of $s_{pos,\,front} = 100\,\mathrm{m}$ and a velocity $s_{vel,\,front} = 16.6\,\mathrm{m/s}$ ahead. Another object (orange) drives on the left lane with a velocity of $s_{vel,\,left} = 50\,\mathrm{m/s}$ and a longitudinal distance of $s_{pos,\,left} = -80\,\mathrm{m}$ behind the ego vehicle.	No lane change	✓
TC 23)	 The ego vehicle (blue) drives at $s_{vel,\,ego} = 33.3\,\mathrm{m/s}$ on the center lane of a three lane highway with a speed limit of $s_{vel,\,speedlimit} = 33.3\,\mathrm{m/s}$ in a country with a right lane driving order.	Lane change right	✓
TC 27)	 The ego vehicle (blue) drives at $s_{vel,\,ego} = 25\,\mathrm{m/s}$ on the center lane of a three lane highway with a speed limit of $s_{vel,\,speedlimit} = 33.3\,\mathrm{m/s}$. A slower object (green) with a velocity of $s_{vel,\,front} = 25\,\mathrm{m/s}$ drives $s_{pos,\,front} = 50\,\mathrm{m}$ in front of the ego vehicle. Another object (orange) drives on the left lane $s_{pos,\,left} = -35\,\mathrm{m}$ behind the ego vehicle with a velocity of $s_{vel,\,left} = 26.38\,\mathrm{m/s}$.	Lane change left	✓

one module like the tactical behavior planning module but rather any subset of modules of the automated vehicle as a whole. Figure 13.4 illustrates a scenario-based closed-loop testing of the feature extraction and model-based filtering, context modeling, guidance, and stabilization.

With the scenario-based closed-loop testing, the interaction of ideally the whole chain can be tested. The control and behavior response from the driving function is used to influence future scenes and by this – implicitly – future situations.

As illustrated in Figure 13.2, scenes will be modified over the course of the scenario according to prediction models in the test system. Thus, this mode of testing makes it possible to test model-based filtering approaches. Other than in the situation-based open-loop testing, objects may move ahead and initiate maneuvers over the course of the simulation time. Such maneuvers may either be triggered by driver models in the test system or defined externally by events in the scenario.

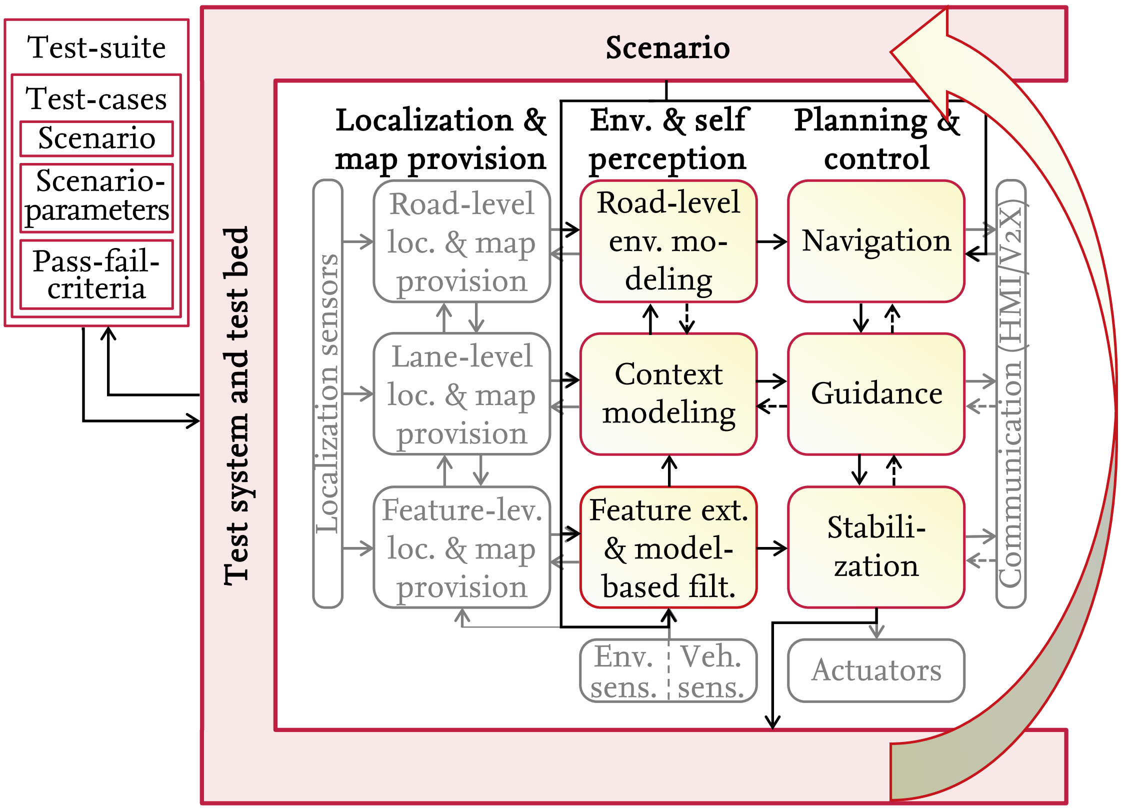


Figure 13.4: Example of scenario-based closed-loop testing of perception and planning & control modules (env. = environment, ext. = extraction, filt. = filtering, lev. = level, loc. = localization, sens. = sensors, veh. = vehicle)

Strengths

The strengths of scenario-based closed-loop testing are short development cycles for driving function development. In fact, a certain complexity level of scenarios can only be tested in a simulation framework in a resource efficient manner. This test technique permits the creation of scenarios to match the particular needs of a developer within minutes. The closed-loop test allows testing of the interaction of several modules and helps to identify and analyze signal latencies or functional instabilities.

The author sees potential in using scenario-based closed-loop testing to obtain meaningful test results for system validation, but proof of this is yet to be provided.

Among the limitations of scenario-based closed-loop testing is the representativeness of the results. E.g., the simulation is based on several models of the real world; starting with behavior models, vehicle dynamic models, sensor models, and even the design patterns of the scenery itself. The big caveat of such a simulation-based design process is that a solution may be particularly well-tailored to a simulation environment but not necessarily to the challenges in the real world. At the time of writing, the modeling of perception-induced measurement errors and uncertainties is not yet sufficiently close to the real issues. While there are certain sensor models to emulate particular artifacts of sensor systems, today's error models still lack the modeling of high-level errors such as false classifications, false segmentations, model-incompliant movement behavior or false semantic associations between perceived entities. Moreover, a sophisticated simulation tool chain comes with significant overall complexity. Therefore, it often takes a lot of time to achieve the intended results.

Limitations

For the scenario-based closed-loop tests, Vires' Virtual Test Drive (VTD)[3] is used as a test system. It is a simulation tool chain for road traffic, railroad, and flight simulation. It provides tools for creating road networks and scenarios as well as a simulation backbone and rendering tools for the visualization. Moreover, it encompasses several simulation models for vehicle dynamics, driver behavior, and pedestrians. As a test bed, an ADTF filter graph is used to translate between VTD's simulation interfaces and the appropriate interfaces of the driving function.

Virtual Test Drive

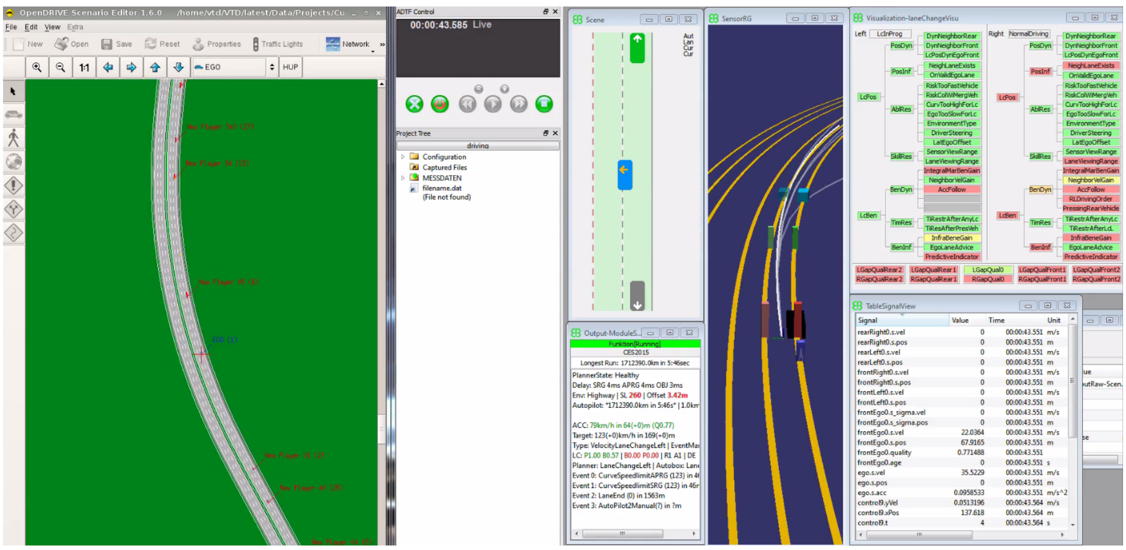


Figure 13.5: Scenario-based closed-loop testing in Virtual Test Drive (VTD) and ADTF

Figure 13.5 illustrates scenario-based closed-loop testing showing the same scene of a scenario side by side in VTD's scenario editor and the ADTF driving function filter graph with an enabled visualization.

[3]Vires Virtual Test Drive, www.vires.com visited on 05/02/2016.

Scenario Description

In Figure 13.6, an exemplary highway scenario for lane changes is depicted. Initially, the automated vehicle drives at 35 m/s on the rightmost lane of a highway. It is approaching a slower vehicle in front of it. The dynamic disadvantage of following a slower vehicle motivates the automated vehicle to perform a lane change to the left (cf. subfigure (1) in Figure 13.6). Since there is no slow vehicle in front to follow, the automated vehicle accelerates to reach the target velocity of $s_{vel,\,target} = 35\,\mathrm{m/s}$.

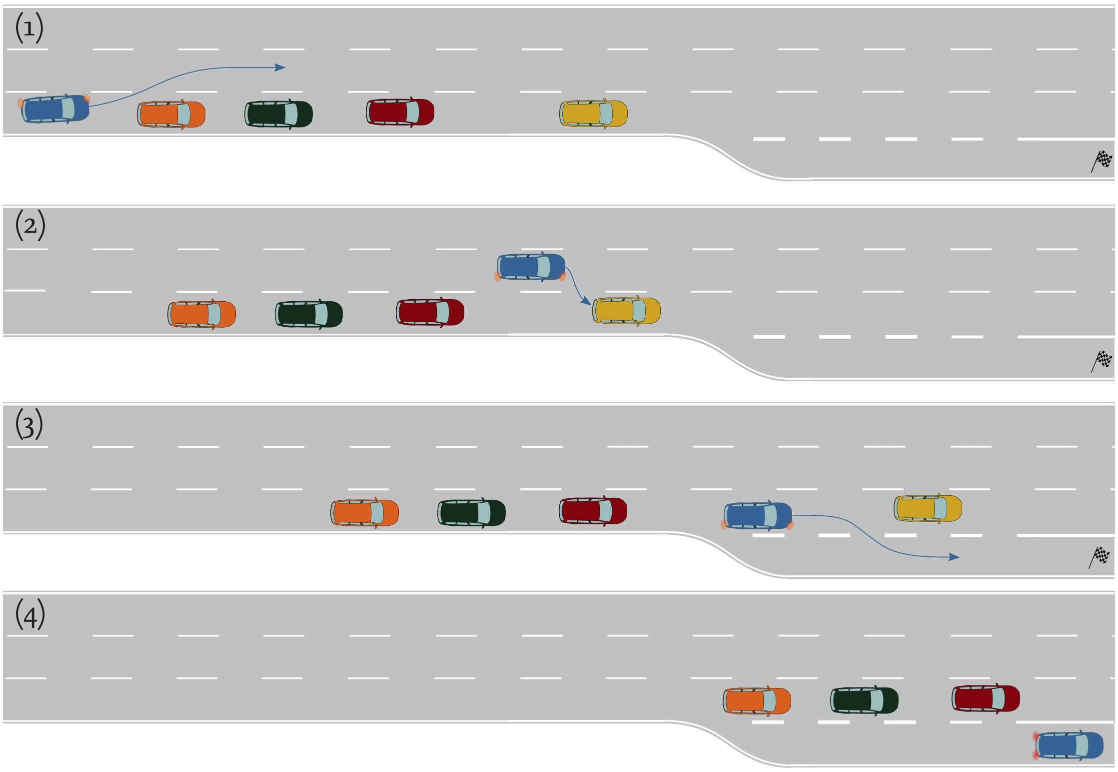


Figure 13.6: Illustration of a highway simulation scenario to test a lane change left due to (1) dynamic benefits, (2) longitudinal gap adjustment, (3) a lane change right, and (4) exiting a highway to an exit ramp for an automated vehicle (blue)

After passing several slower vehicles on the right lane by driving on the center lane of a three lane highway, the automated vehicle gets close to an exit where it is commanded to leave the highway. Given the relative speed difference, the gaps on the right lane are too small for a direct lane change. Hence, the automated vehicle initiates a longitudinal gap adjustment to center itself to the best reachable and best sized gap. To achieve this, the automated vehicle decelerates to reduce the speed difference between itself and the traffic on the right target lane (gap adjustment). This is followed by a lane change to the right into the selected gap (cf. subfigure (2) in Figure 13.6). Last of all, the automated vehicle exits the highway by changing once more to the right on a deceleration lane of the targeted highway exit (cf. subfigure (3) in Figure 13.6). The automated vehicle adapts its speed to the upcoming curvature of the exit ramp (cf. subfigure (4) in Figure 13.6).

Explanation of Test Results

Figure 13.7 illustrates selected state variables that are relevant for the lane change behavior planning in the aforementioned scenario. The first three figures depict relevant state variables of the situation the automated vehicle is facing: namely, the (automated) ego vehicle's velocity longitudinally to the lane, the lateral offset to the center of the ego lane, and the distances to other objects directly in front of the automated vehicle in the ego lane and the right neighbor lane.

Moreover, Figure 13.7 shows some hidden state variables resulting from a situation assessment (cf. section 10.3). The two most relevant as depicted here are the lane change possible estimate and the lane change beneficial estimate. Last of all, Figure 13.7 shows the lane change action resulting from the overall lane change planning process.

Lane Change Left

The previously described scenario can be traced in the measured data. The longitudinal ego velocity plot visualizes the aforementioned velocity profile. Initially, the automated vehicle drives at $s_{vel,\,ego} = 35\,\mathrm{m/s}$. After experiencing a marginal slow down due to a vehicle in front of it, it activates the indicator to the left (cf. *LcState* = *IndicateLcLeft* in the last subfigure of Figure 13.7) and executes a lane change by building up a lateral offset to the left (cf. $d_{ego,\,pos} = 1.8\,\mathrm{m}$; subfigure (1) in Figure 13.6). After crossing the left lane boundary, the lane detection changes the ego lane from the highway's rightmost lane to the center lane of the three lane highway. As the reference lane switches it causes a jump in the lateral ego lane offset from $d_{ego,\,pos} = 1.8\,\mathrm{m}$ to $d_{ego,\,pos} = -1.8\,\mathrm{m}$. After this jump, the automated vehicle re-centers itself to the new lane. On this middle lane of the highway there are no other vehicles closely in front of the automated vehicle. Hence, it accelerates to reach the target velocity $s_{vel,\,target} = 35\,\mathrm{m/s}$.

Several vehicles are passed or overtaken. This is illustrated by the distance of the immediate next vehicle in the ego lane and the right neighbor lane. Several slower vehicles are approached from behind (decreasing distance) until they are overtaken. In total, ten vehicles are overtaken by the automated vehicle.

Longitudinal Gap Adjustment

After a certain period of driving the scenario requires the automated vehicle to exit the highway at an upcoming highway exit on the right. Therefore, the automated vehicle is required to change back to the rightmost lane of the highway.[4] To achieve this, the automated vehicle activates a longitudinal gap adjustment (cf. section 10.3.3) at $t = 78\,\mathrm{s}$ (cf. *LcState* = *PrepareLcRight* in the last subfigure of Figure 13.7; subfigure (2) in Figure 13.6). To simplify a lane change, the relative velocity difference between the automated vehicle ($s_{vel,\,ego} = 35\,\mathrm{m/s}$) and the objects/gaps on the right lane ($s_{vel,\,target} = 22\,\mathrm{m/s}$) has to be reduced. After a few seconds, a lane change becomes possible and the automated vehicle changes to the rightmost lane of the highway at $t = 90\,\mathrm{s}$ by activating the indicator right and building up a lateral displacement to the right (cf. *LcState* = *DoLcRight*). At $t = 95\,\mathrm{s}$ the lane change is finished and the automated vehicle is fully re-centered to the rightmost lane. After a few more seconds it reaches the beginning of the exit ramp (cf. subfigure (3) in Figure 13.6). Even before the exit ramp is actually next to the automated vehicle, the right indicator is activated based on the information from an a-priori map that the exit ramp is about to appear on the right.

Exiting the Highway

[4]The urge to change back to the rightmost lane is derived from the lane advice (cf. section 10.3.1). In this simulation, the automated vehicle changes far later than it typically does on a regular road. This is due to an unusually late change of the lane advice, which is caused by longer than usual lane segments in the simulated road network.

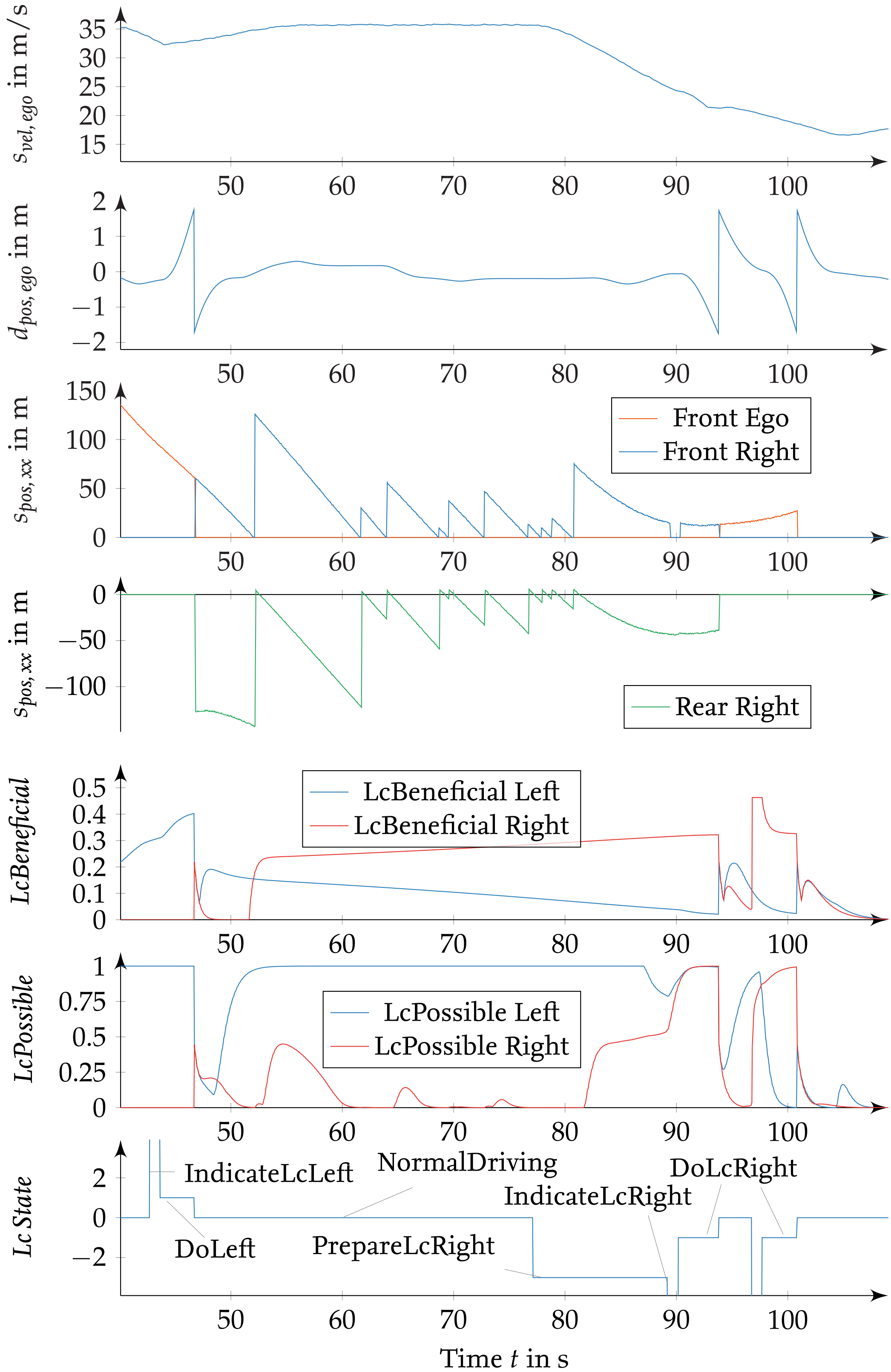


Figure 13.7: Measured data for a scenario-based closed-loop testing scenario on a highway. $s_{vel,\,ego}$ is the ego velocity; $d_{pos,\,ego}$ is the lateral offset of the ego vehicle to the lane center; $s_{pos,\,xx}$ is the longitudinal distance towards the front ego (FE), front right (FR), or rear right (RR) vehicle (cf. attachment D); *LcBeneficial* and *LcPossible* = situation assessment, whether lane change is beneficial/possible; *LcState* = integer value if lane change is prepared, indicated, or executed, or regular driving in lane is performed

This predictive indicator mode (cf. section 10.5.2) helps to prevent conflicts with other vehicles changing faster to the exit lane and possibly blocking the automated vehicle from being able to change to the exit ramp. After the exit ramp has reached its full width, the automated vehicle changes onto the exit ramp and slowly adapts its speed (cf. $s_{vel,\,ego} = 18\,\mathrm{m/s}$; subfigure (4) in Figure 13.6) to follow the right turn of the exit ramp.

Predictive Indicator Mode

13.4 Conclusions

In this chapter, situation-based open-loop testing and scenario-based closed-loop testing has been introduced as a concept and demonstrated for testing lane change behavior planning. Different levels of testing that proved to be useful for the testing of driving function modules are: unit tests, situation-based open-loop tests, scenario-based closed-loop tests, and real world driving tests as different steps for system testing and validation. The different advantages and limitations of the test methods are introduced.

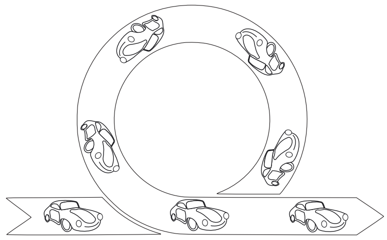
The chapter does not address a systematic approach to generate test-cases as in Schuldt (2017). So far, they were manually designed by a human expert. Moreover, the test-case evaluation is not automated for the scenario-based closed-loop simulation. At the moment a human expert needs to define and evaluate a pass-fail criteria for each test-case. While this works well for testing during the development phase, it scales unfavorably for validation tests with thousands or millions of test-cases automatically executed on a simulation server farm.

Open Issues

The next step is to use the situation-based open-loop and scenario-based closed-loop test methods combined with systematic test-case generation to automatically find system boundaries and scenarios where the system under test shows a lower performance. This includes an automatic test-case evaluation based on eligible pass-fail criteria.

Next Steps

14 Maneuver-Based Performance Evaluation in Real Traffic



Real world driving tests are the last step in the test process presented in Chapter 13. They are used for system and acceptance tests in the V-Model and simulation model verification. Necessitating a real vehicle or prototype, they should be executed after successfully passing all situation-based open-loop and scenario-based closed-loop tests (cf. Chapter 13). [1]

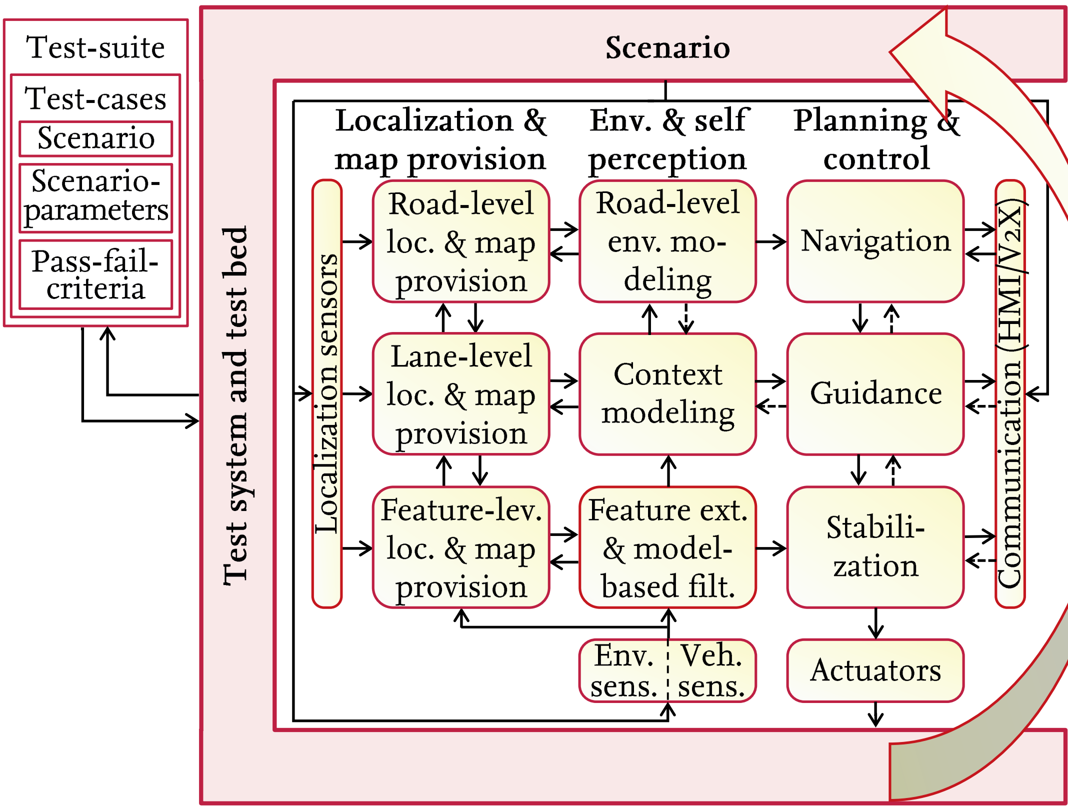


Figure 14.1: Real world driving testing (env. = environment, ext. = extraction, filt. = filtering, lev. = level, loc. = localization, sens. = sensors, veh. = vehicle)

A test of driving functions with real environment conditions and resulting uncertainties is currently only possible in real world driving tests. Real world driving tests also require test-cases with scenarios, scenario-parameters, and pass-fail-criteria (cf. section 2.3). The scenario parameters are given indirectly, e.g., by actually driven trajectories or a hereof resulting physical arrangement of other traffic participants.

[1]The introduction of this chapter has been pre-published by the author in Ulbrich et al. (2017c).

Figure 14.1 illustrates the modules in a system architecture (cf. Chapter 3), which can be tested with real world driving tests.

Advantages

The advantage of real world driving tests is that they do not require simulation models and thus do not induce errors by incorrect models. Real world driving tests are to some extent random tests. Thus, many different scenarios will automatically be tested without a test-case generation. Hence, driving tests in real traffic will by default cover situations which have never been thought of at design time.

Limitations

The aspect of – to some extent – random testing simultaneously imposes a severe limitation of the test method: Due to the random behavior of other traffic participants, the test-cases are not fully reproducible. Moreover, if a scenario is not specifically scripted, rarely occurring events may require millions of driving kilometers before they can be tested. Finally, the test-cases cannot be performed faster than in real-time. Because of this, executing a large number of test-cases is directly linked to high costs and expenditure of time.

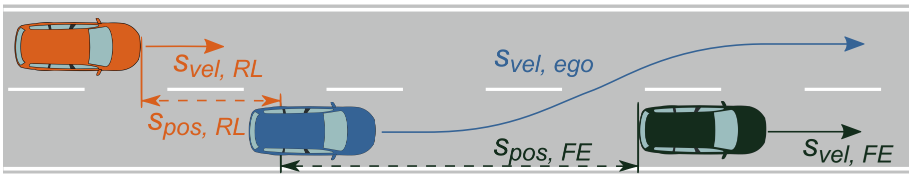


Figure 14.2: Scenario used for evaluation (for coordinate system cf. attachment D)

Critical Consideration

Often a conference or journal publication ends its evaluation by successfully demonstrating one lane change maneuver. Yet, successfully completing one maneuver is a necessary but insufficient criterion for the eligibility of using an algorithm in regular traffic. One could even ask whether that is truly an evaluation if the only evaluation criterion is whether one demonstration failed or not. Here, a lack of true metrics on a behavioral level becomes evidently visible for individual maneuvers. Better metrics exist on a more aggregated level as in Chapter 15, but it is still necessary to understand how behavioral planning is done for a single maneuver before it makes sense to aggregate individual maneuvers onto a macroscopic level.

It is neither possible nor useful for this thesis to evaluate every distinct situation an automated vehicle might face in real world traffic. Thus, the remainder of this section will show the evaluation of variations of one key scenario as in Figure 14.2 and traversing a cloverleaf highway interchange. In section 14.1 the author tests the scenario where the yellow car is slower than the ego vehicle and in section 14.2 it is assumed that the yellow car is faster than the ego vehicle.

14.1 Successfully Completing a Lane Change [2]

Microscopic Analysis of a Lane Change

In this section, single lane change maneuvers are analyzed in detail. Figure 14.2 presents a scenario for a lane change on a German highway. A slower front vehicle in the ego lane is overtaken by a lane change to the left. Later on, the automated vehicle obeys the right lane driving order on German highways and changes back to the right. Figure 14.3 illustrates the scenario with a sequence of camera images.

[2] This subchapter has been pre-published by the author in Ulbrich & Maurer (2015a).

Figure 14.4 presents different state variable estimates during the course of the scenario. It depicts the lateral offset of the automated vehicle to the center of the lane $d_{pos,\,ego}$, the ego velocity $s_{vel,\,ego}$, and the distance to the leading front vehicle.

In Figure 14.4 the first two plots depict driving a 20 km stretch of the A9 from Ingolstadt, Germany northbound. The longitudinal ego velocity of the automated vehicle is depicted in the first diagram. There is no speed limit on this stretch of a $3+3$ lane highway. The target velocity is set to 40 m/s. Occasionally traffic in front of the automated vehicle slows it down if it is not able to perform a lane change due to traffic on the neighbor lanes.

The lateral offset of the automated vehicle to the center of the ego lane is shown in the second and third subfigures of Figure 14.4. Each time a lane change is executed, the ego lane jumps to another lane and the lateral offset jumps from negative to positive (lane change right) or positive to negative (lane change left). The third to eighth subfigures of Figure 14.4 illustrate the situation assessment before and during a lane change to the left.

The maneuver is visualized by a sequence of images from the lane tracking camera and a situation visualization widget in Figure 14.3. Initially, the automated vehicle drives on the middle lane of a three lane highway. In front of it a slow truck (green) appears. As overtaking on a highway in Germany is only allowed on the left, a situation assessment for a lane change to the left is evaluated. The fourth to sixth plots in Figure 14.3 illustrate the distances and velocities of the immediate next vehicles on the left neighbor lane and on the ego lane in front. Starting from $t = 442$ s a slower vehicle with a velocity of about 33 m/s is detected 150 m in front. For the first few seconds a lane change to the left is still considered possible to some extent at the current time step (cf. *Lane change possible left*) in the eighth plot. However, due to the planning ahead into the future and due to a still relatively low disadvantage of staying behind a vehicle that is 150 m away, no lane change is executed. To $t = 463$ s the rear left neighbor lane's vehicle approaches the automated vehicle from behind until it is next to the automated vehicle. Here, the blindspot radar sensors do not allow an accurate position estimation but only an object existence estimation. At $t = 463$ s the same object is seen again by the front laser scanner and the distance towards this object increases again.

A second vehicle approaching the automated vehicle on the left neighbor lane is detected -30 m behind with a velocity of 44 m/s at $t = 465$ s. This has passed the automated vehicle at $t = 470$ s. Behind this fast vehicle no other vehicle follows. Hence, a gap opens up to the left. In the meantime, the automated vehicle has approached the slow vehicle in front of it on the ego lane to a relative distance of 42 m. To avoid a collision, the automated vehicle decreases its speed to 33 m/s. Therefore, it obtains a high dynamic benefit of performing a lane change to the left to reach its target velocity of 40 m/s again (cf. *Lane change beneficial left* in the seventh plot). The lane change decision making module decides to activate the indicator at $t = 474$ s (*Lane change state* changes to $+9$). Later on, at $t = 475$ s, it initiates a lane change to the left (*Lane change state*$= +1$). This can be seen by the lateral offset to the center of the ego lane $d_{pos,\,ego}$ increasing until the lane markings have been passed and the automated vehicle re-centers to the left neighbor lane.

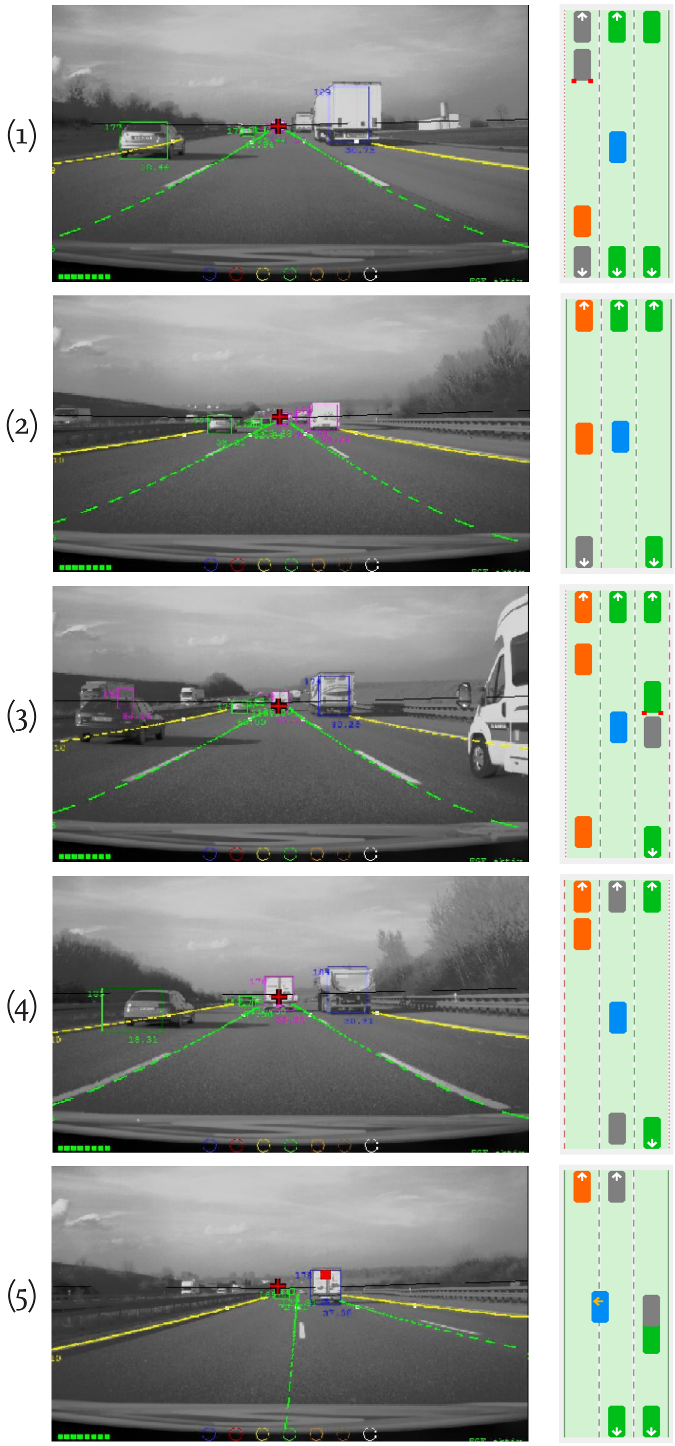


Figure 14.3: Video and situation visualization for lane change situation assessment. Ego vehicle (blue); white arrows indicate squeezed vehicle distances

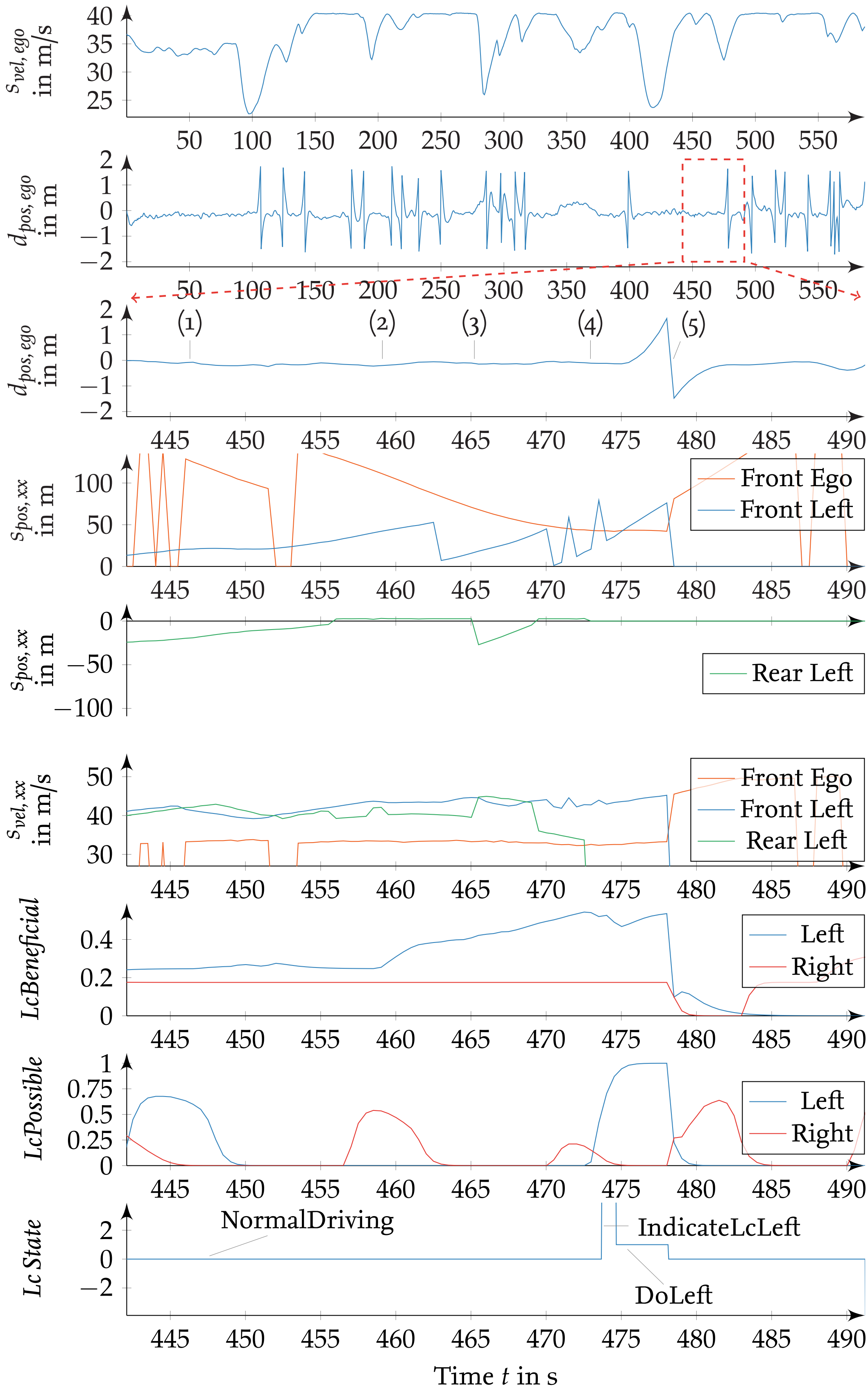


Figure 14.4: State variables during an overtaking scenario. $s_{vel,ego}$ is the ego velocity; $d_{pos,ego}$ is the lateral offset of the ego vehicle to the lane center; $s_{pos,xx}$ is the longitudinal distance towards the front ego (FE), front left (FL), or rear left (RL) vehicle (cf. attachment D); $s_{vel,xx}$ are the velocities of these vehicles; *LcBeneficial* and *LcPossible* = situation assessment, whether lane change is beneficial/possible; *LcState* = integer value if lane change is indicated or executed, or regular driving in lane is performed

14.2 Aborting a Lane Change

For safe execution of any maneuver it is essential to have the option of aborting that maneuver. This section illustrates the abortion of a lane change. It illustrates the abortion of a lane change to a left neighbor lane due to a fast vehicle from behind.

Necessary Viewing Ranges

No lane change abortions would be necessary if the sensor viewing range to the rear would be sufficient to detect fast approaching vehicles early enough that either no lane change were initiated or could have been completed before a fast rear vehicle had reached the ego vehicle. This aspect has been translated into a formula in section 10.3.2. It has been evaluated for different relative velocities, required time gaps, and assumed braking accelerations in attachment E. With today's sensor viewing ranges of maybe 70 m to 150 m, no safe lane change would be possible on a German highway with no speed limit. For that purpose, an ability restriction has been incorporated into the system in section 10.3.2. This mechanism prevents lane changes if there is no speed limit and the automated vehicle wants to change to the leftmost lane of a highway with no sufficiently slow vehicle in sight. It has been disabled to simplify this analysis of aborting a lane change.

Deactivated Ability Restriction

Figure 14.5 illustrates the scenario using a sequence of images. At the very left of each snapshot the video of the front camera is shown. In the center, a graph-based scene visualization illustrates the perceived environment of the automated vehicle. The very right part of each snapshot displays a situation visualization. The distances in the situation visualization are compressed. If a vehicle is further away than it is representable in the situation visualization, an arrow pointing away is added to the shape of the vehicle. Figure 14.6 depicts relevant state variables for the lane change abortion.

Indicator Activation

Before $t = 26$ s, a lane change to the left neighbor lane is not possible due to a faster, overtaking vehicle on the left neighbor lane. Initially, it is behind the automated vehicle and passes it at $t = 24$ s. The vehicle in front of the automated vehicle drives slower than the target velocity, hence overtaking it is beneficial and therefore the automated vehicle starts flashing the indicator left at $t = 26.7$ s. At $t = 27.6$ s, another vehicle is detected on the left neighbor lane but with a velocity of only 35.45 m/s it is sufficiently slow to complete the lane change. Thus, at $t = 27.7$ s the automated vehicle starts to build up a lateral offset $d_{pos,ego}$. However, before completing the lane change, it is aborted at $t = 29.1$ s. This is due to detecting a fast rear neighbor vehicle with a velocity of 52 m/s at an initial distance of 110 m to the rear. This vehicle was out of the sensor viewing range when the lane change was initiated.

Fast Rear Neighbor Vehicle

Not aborting the lane change would have necessitated a strong deceleration of the fast vehicle with $-3.5\,\mathrm{m/s^2}$ to maintain a time gap of 0.8 s while ignoring any reaction time (cf. section 10.3.2). Due to the inertia of mass of the automated vehicle, the lateral offset $d_{pos,ego}$ does not instantaneously decrease but continues to build up to 0.245 m at $t = 30.4$ s. At $t = 34$ s the fast rear vehicle overtakes the automated vehicle.

The evaluation demonstrates the successful handling of a trade-off situation. The value dimension (cf. Chapter 9) of *creating user satisfaction* is traded off against *creating third party satisfaction* or even the *protection of the physical integrity of things* in case the vehicle approaching the automated vehicle from behind were not able to avoid a collision (cf. Figure 9.1). The utility to perform a lane change *due to the dynamic traffic situation* (cf. Figure 9.4) is balanced against the risk from a collision *caused by the non-mastery of the driving situation* (cf. Figure 9.3) of initiating a lane change at a time, when the fast approaching rear vehicle was not yet detected.

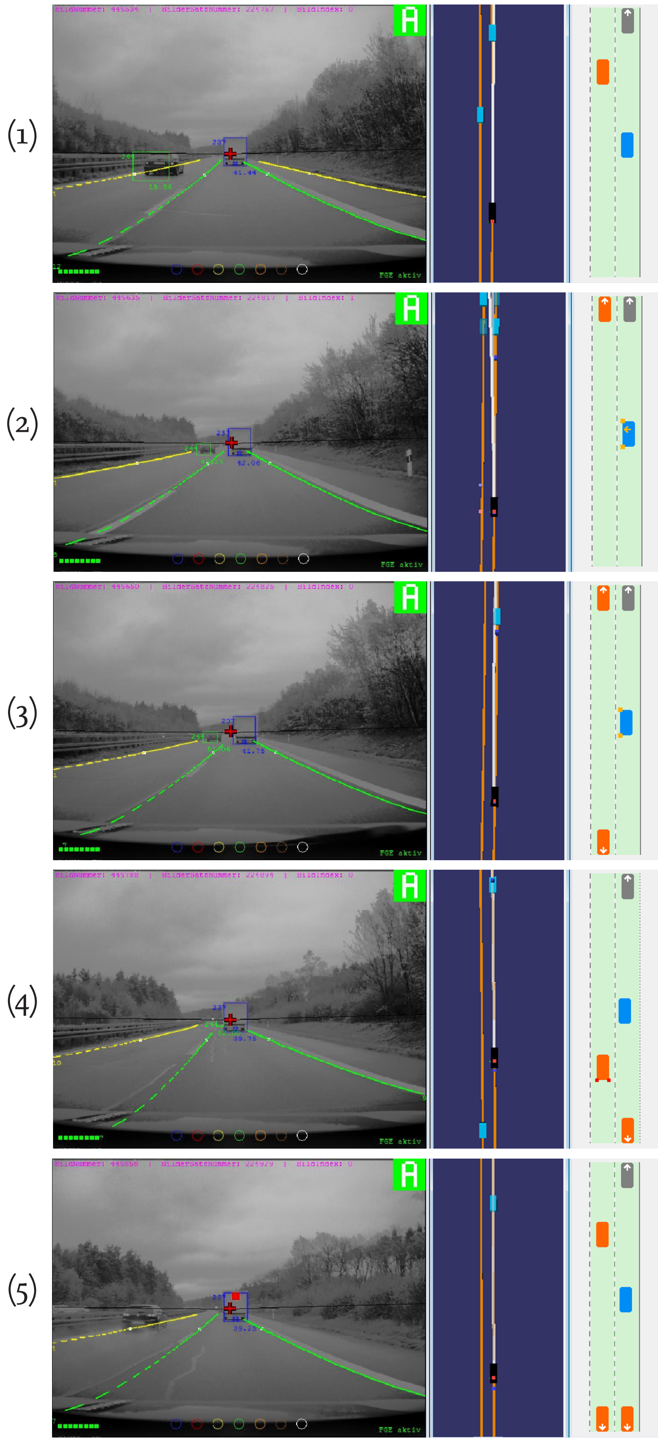


Figure 14.5: Video and situation visualization for a lane change abortion. Ego vehicle (blue) attempts to overtake slightly slower front vehicle (grey). Faster vehicles (red) on left lane necessitate abortion of a lane change. White arrows indicate squeezed vehicle distances

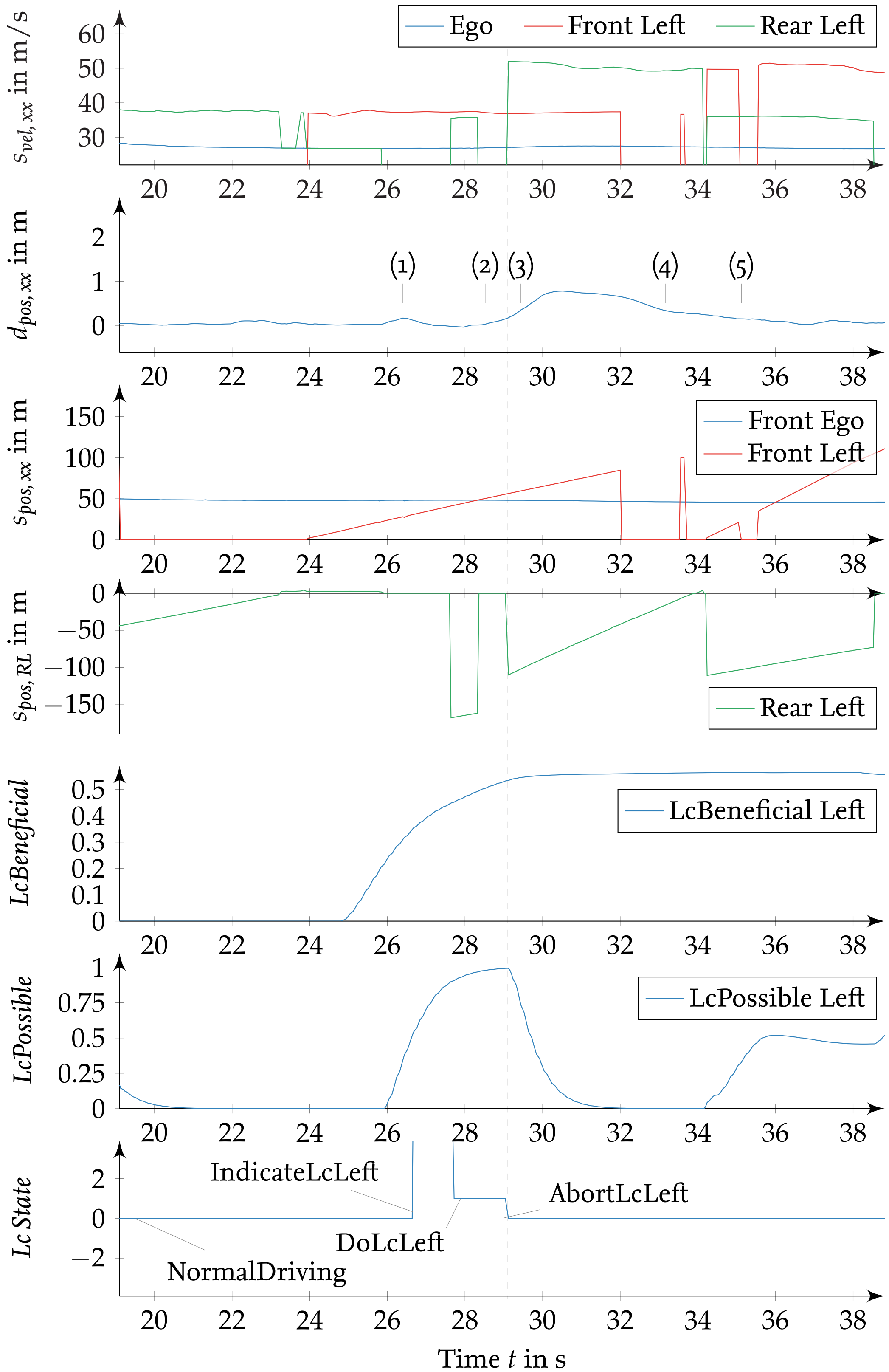


Figure 14.6: State variables during the abortion of a lane change to the left. $s_{vel,ego}$ is the ego velocity; $d_{pos,ego}$ is the lateral offset of the ego vehicle to the lane center; $s_{pos,xx}$ is the longitudinal distance towards the front ego (FE), front left (FL), or rear left (RL) vehicle; $s_{vel,xx}$ are the velocities of these vehicles (cf. attachment D); *LcBeneficial* = situation assessment, whether lane change is beneficial; *LcPossible* = situation assessment, whether lane change is possible; *LcState* = integer value for state of lane change process, if lane change is indicated, executed, aborted, or regular driving in lane is performed

14.3 Passing a Highway Interchange

Interchange Scenario

So far, the analysis has focused on lane changes in the regular traffic flow of a highway. For a technical system, these are far easier than infrastructure restricted lane changes in highway interchanges because maneuvers are far less time critical and thus easier gaps may be selected for lane changes. Figure 14.7 presents a cloverleaf highway interchange between the German highway A2 and A391 close to Braunschweig. In total the interchange requires exiting the A2 to an off-ramp to a distribution lane, to change to an indirect connection ramp in a weaving area, to drive on an indirect connection ramp with a high curvature to merge back into a distribution lane in a second weaving area and finally to merge back into the traffic at the A391 highway at the on-ramp. Thus, in total the highway interchange requires at least four lane changes plus possibly additional lane changes to get to the off-ramp and to change back to a not rightmost lane on the destination highway.

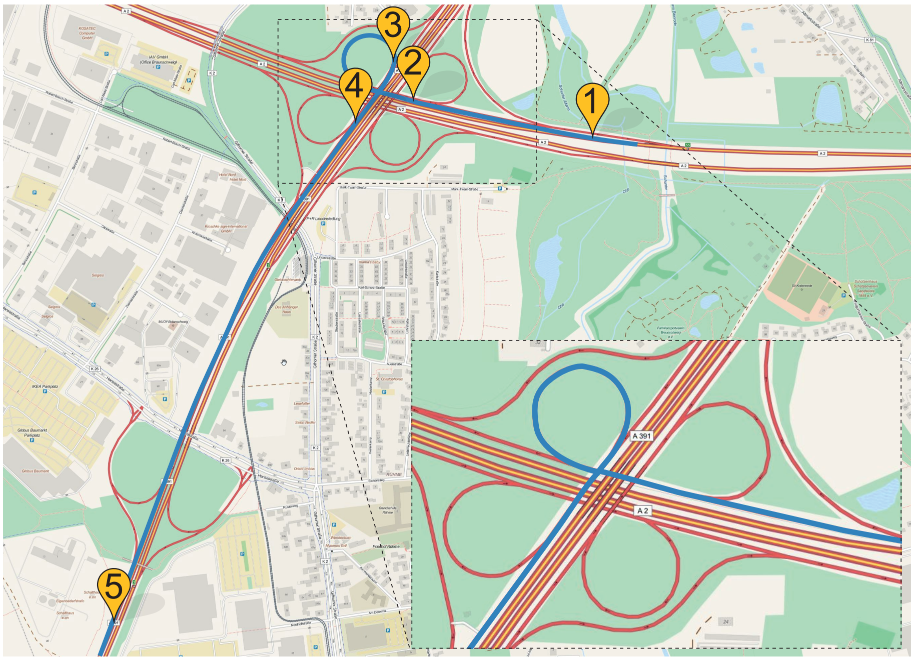


Figure 14.7: Map of cloverleaf highway interchange between A2 and A391 in Germany. Snapshots from Figure 14.8 highlighted in red. Source: OpenStreetMap

Key Scenes

Five key scenes from Figure 14.7 are highlighted with red pinpoints and details for these are provided in Figure 14.8. It illustrates the scene with a screenshot from a video camera, a graph-based environment model as well as a simplified situation-visualization.

Figure 14.9 presents different state variables in the course of the scenario in Figure 14.7. It depicts the lateral offset to the center of the ego lane, the discrete lane change status the system is in, and the lane change possible as well as lane change beneficial estimates.

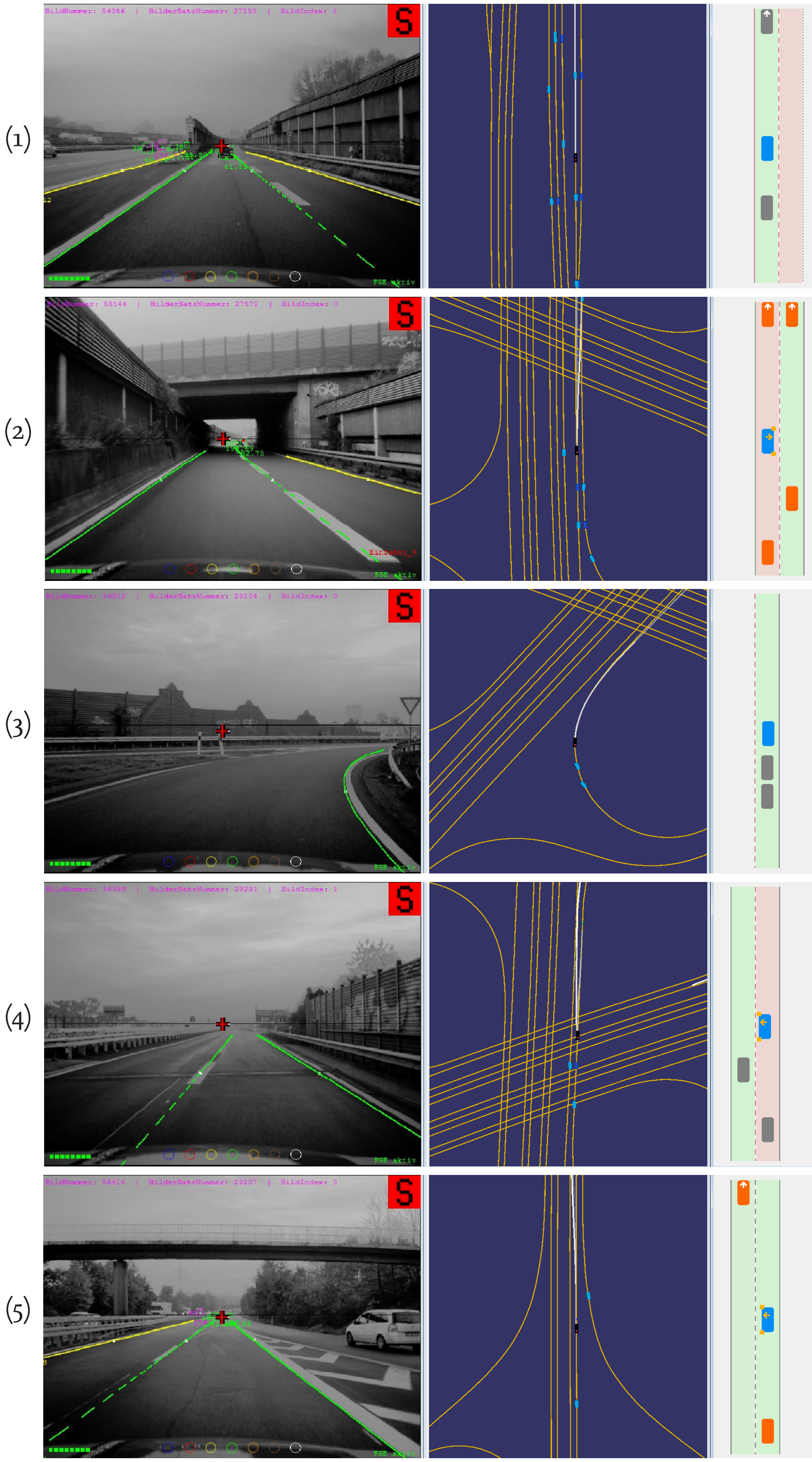


Figure 14.8: Video, scene, and situation visualization while passing a highway interchange. Ego vehicle (blue); faster vehicles (red); vehicles with ego speed (grey); white arrows indicate squeezed vehicle distances; red/green lane color indicates low/high lane advice to reach navigation destination

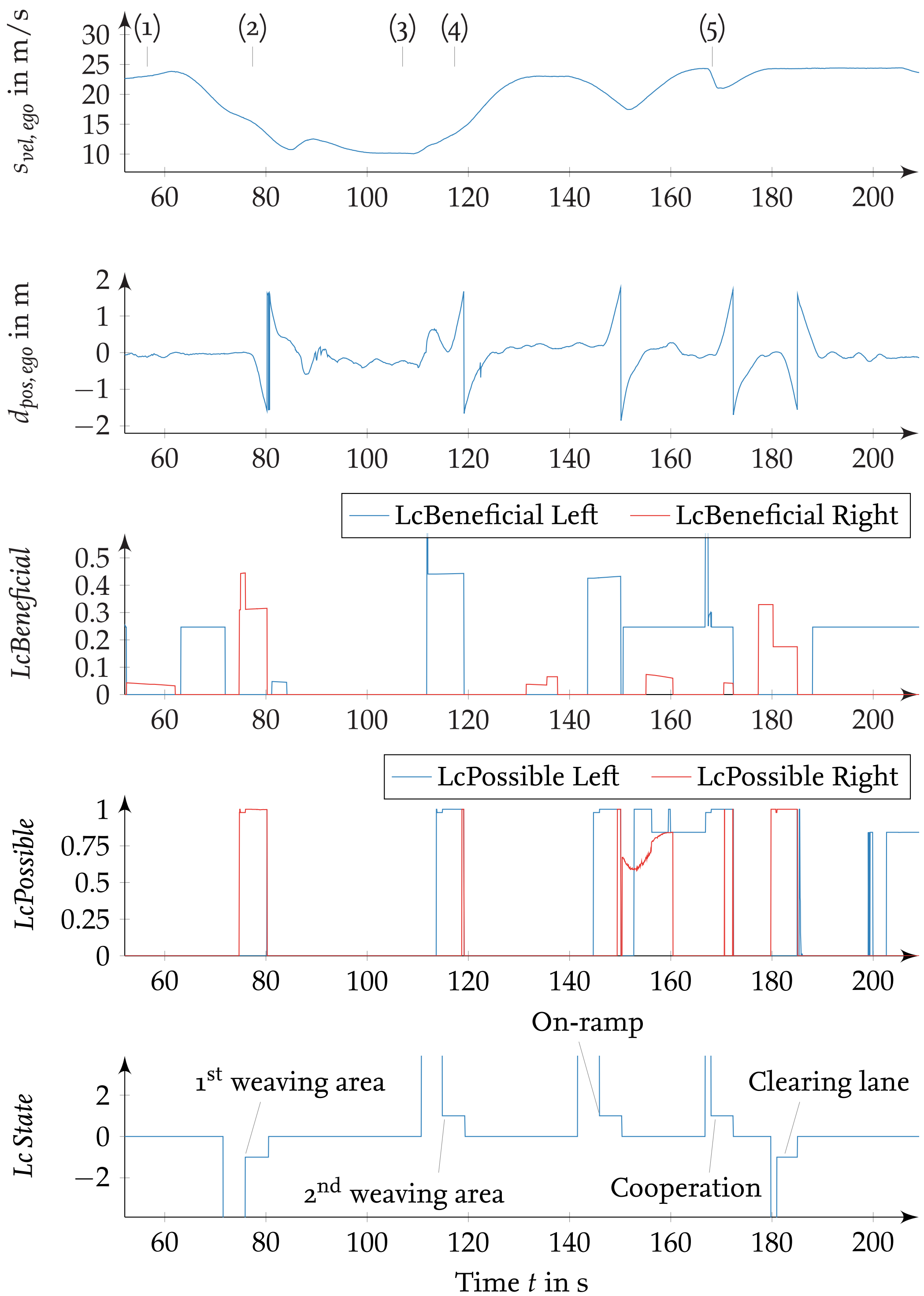


Figure 14.9: State variables during a highway interchange scenario. $s_{vel,ego}$ is the ego velocity; $d_{pos,ego}$ is the lateral offset of the ego vehicle to the lane center (cf. attachment D); *LcBeneficial* = situation assessment, whether lane change is beneficial; *LcPossible* = situation assessment, whether lane change is possible; *LcState* = integer value for state of lane change process, if lane change is indicated, executed, or regular driving in lane is performed

Highway Interchange Process in Detail

The automated vehicle starts off at the off-ramp of the A2 highway at $t = 52\,\mathrm{s}$. It proceeds through the highway interchange on a distribution lane until it reaches the first weaving area where it has to change to the 270° indirect ramp. The indicator to the right is activated at $t = 71.6\,\mathrm{s}$ even before the right neighbor lane is perceived based on an a-priori map (predictive indicator activation). At $t = 75.9\,\mathrm{s}$ the automated vehicle performs a lane change to the 270° indirect ramp. At the time of writing, the automated vehicle drives slower than a human driver on the 270° indirect ramp to enhance lane tracking and lateral control. At $t = 100.8\,\mathrm{s}$ once again a predictive indicator activation is performed to prepare a lane change to the left at $t = 114.9\,\mathrm{s}$ in the second weaving area parallel to the A391 highway. The automated vehicle traverses on the distribution lane until it reaches the on-ramp to the A391 at $t = 141.6\,\mathrm{s}$. It changes back to the A391 highway at $t = 145.9\,\mathrm{s}$. Driving on the rightmost lane of the A391 highway it reaches the next highway on-ramp at $t = 166.8\,\mathrm{s}$. As illustrated in (5) of Figure 14.8, another vehicle is entering the A391 highway at the on-ramp. As described in section 4, the automated vehicle cooperatively clears the rightmost lane of the highway to simplify merging for the vehicle on the on-ramp. At $t = 181.0\,\mathrm{s}$ it changes back to the rightmost lane to follow the right lane driving order in Germany and clears the left lane for a faster vehicle from behind.

Challenges

At the time of writing it is still very challenging to pass an entire highway interchange without human interaction. Challenges are manifold: To decide reasonable lane change maneuvers, it is necessary to locate the automated vehicle correctly on lanes within the highway interchange. While this is challenging on open roads, it is even more challenging with tunnels, bridges, and noise protection walls directly next to the lanes. Perceiving and tracking lane markings is far more challenging in highway interchanges than on regular roads due to a more complex geometry, tar joints at bridges, etc. Thus, it is often difficult to determine the type of lane markings, not only for the ego lane but also for the neighbor lanes necessary for map-related localization and behavior planning for line crossing. Within a highway interchange high lane curvatures in a non-flat road topology need to be detected correctly, otherwise the automated vehicle would miss the beginning of ramps or would weave around within a lane. This is particularly noticeable on 270° indirect ramps between two highways. Incorrectly identifying the type of lane boundaries might result in delayed lane changes within weaving areas. Given that weaving areas are quite limited in length, delays might mean an exit is missed.

Perception Issues

Furthermore, delayed lane changes in weaving areas may result in an impatient driver of a vehicle behind the automated vehicle overtaking or at least tailgating the automated vehicle in the weaving area and thus making lane changes even more challenging. Tailgating with a lateral offset could result in a rear vehicle leaving the sensor viewing range (cf. attachment B) of the rear Lidar sensor. Thus, it may only be seen by the radar sensors covering the blind spot area at the side of the automated vehicle and thus not be perfectly locatable and hence may block a lane change.

The radar sensors covering the side areas (cf. attachment B) impose a second limitation: Due to narrower lanes, it may be that lateral distances are incorrectly detected and thus, e.g., a guard rail closely behind the second lane in a weaving area is detected as an object blocking that lane in the weaving area.

Maneuver Time

Yet another challenge results from the length of on-ramps and in particular weaving areas: Often they are even relatively short for human drivers. It is necessary to identify merging and cooperative slowdowns early and without errors. The length is often insufficient for hesitation or decision changes. The lane change execution requires more lateral dynamics than on regular highways (cf. *aggressive lane changes* in section 10.5.2). This may induce additional roll and pitch movements, which again makes it harder to detect lanes and lane curvatures correctly. From the author, the two major aspects that require improvements are on the one hand the lane marking/lane geometry recognition and on the other hand a true 360° perception, which allows to perceive dynamic elements accurately even though they are in direct proximity to the ego vehicle (cf. attachment B).

Human as a Fallback Level

After all, an entire maneuver from the very left lane of a three lane highway, to the rightmost lane of the initial highway, to the off-ramp, through a first weaving area, through a 270° indirect ramp, through a second weaving area to a distribution lane, to an on-ramp back onto the rightmost lane of another highway requires six lane changes in a constrained space with close interaction with other vehicles. Not a single one of them may fail to reach the mission destination. At the time of writing this is still very challenging. In good conditions and easy highway interchanges with favorable traffic conditions it will work most of the time. In difficult scenarios it is – at the time of writing – necessary to have a human driver as a fallback level as in partially automated driving. The human driver is required to monitor the automated vehicle and possibly even to support it in steering or maneuver execution. Hence, the goal of SAE level three automation or higher (cf. section 2.1) is currently not achieved in highway interchanges.

14.4 Conclusions

Contribution

This chapter provided a microscopic evaluation of lane change maneuvers for automated driving. It demonstrated the situation assessment and behavior planning for executing as well as aborting a lane change in free flowing traffic. Moreover, it showed behavior planning in a complex highway interchange scenario.

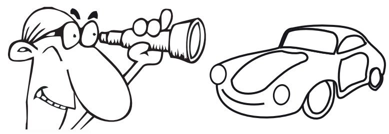
Quantifiability of Behavior

This evaluation proves the feasibility of the concepts, presented in Part II of this thesis, for real world driving. Yet, it lacks clear metrics to quantify *how good* the lane change behavior performance is. This is due to the lack of metrics and possibly even the quantifiability of driving behavior as a whole.

Necessary but not Sufficient

Many conference and journal publications end their evaluation with the successful demonstration of a single lane change maneuver. Yet, successfully completing one maneuver is a necessary but insufficient criterion for the eligibility of using an algorithm in regular traffic. Thus, the next chapter provides a macroscopic evaluation that not only looks at particular maneuvers but rather at the aggregated overall behavior.

15 Macroscopic Evaluation in Real Traffic



To render the evaluation section complete, a macroscopic evaluation in real traffic is essential. Only real traffic exhibits manifold different maneuvers and thus will illustrate the robustness and general applicability of the developed approaches. Changing the focus from individual maneuvers to a macroscopic performance evaluation reflects the reality a human passenger experiences in a vehicle while performing a longer stretch of automated driving.

Evaluation According to Requirements

In the requirements section, it was stated that behavior planning for lane changes should be fast, consistent, provident, deterministic, and in compliance with the system's values. Section 15.1 evaluates the *computational complexity* of the here presented approach. *Consistency* in behavior planning is assessed by analyzing lane change completions and abortions in section 15.2. The situation prediction is a core component for *provident* behavior planning. It is evaluated in section 15.3. The algorithms are by design *deterministic*. No black box models like neuronal networks are used. Thus, no evaluation of the algorithm's determinism was performed.

Compliance with Values

Last of all, behavior planning shall be in compliance with the automated vehicle's system of values. According to Chapter 9, relevant value dimensions are safety, mobility, legality, user satisfaction, and third party satisfaction. As introduced in section 12.3, *safety* is evaluated regarding several aspects. The indicator activation prior to a lane change shall be in a timely manner to inform other drivers as well as the passengers in the vehicle about an intended maneuver (cf. section 15.4.1). Safe lane change execution is evaluated by analyzing the maneuver itself in section 15.4.2. Moreover, the maximally available reaction time during lane changes towards surrounding vehicles is analyzed in section 15.4.3. The challenges to quantify the value dimension of *mobility* have been highlighted in section 12.3. To provide at least some metric, a velocity gain from a lane changes is evaluated in section 15.5. *User satisfaction* is evaluated as part of the subjective assessment in section 15.6. According to the discussion in section 12.3, *legality* and *third party satisfaction* will not be assessed in this thesis.

15.1 Evaluation of the Computational Complexity

The algorithm runs in real-time on an Intel i7 4800MQ CPU sharing resources with trajectory planning, situation modeling, and visualization modules. Typical peak loads for any of the cores are below 20%.

Situation Assessment

Per cycle, the situation assessment is executed several thousand times. This is because according to section 10.2, a tree of future situations is predicted. Observable state variables are predicted according to a car following model for each vehicle and a recalculation of hidden situation aspect estimates is performed by the dynamic

Bayesian network. On average a resulting tree of actions and future situations has 156 situation nodes. An unscented transform with 22+1 sigma points is executed for every situation to estimate the resulting variances in the dynamic Bayesian network. Scene updates from the perception modules are obtained with a rate of 25 Hz. Therefore, the dynamic Bayesian network is executed on average $(156 + 23) \cdot 25 = 4475$ times per second. As there are no big *for*-loops in the dynamic Bayesian network implementation, most evaluation time is used for floating point operations and some if conditions.

Overall Behavior Planning

Considering not only the (repetitive) situation assessment but also the behavior planning as a whole, approximately 50 paths in the tree are evaluated on average in about 5 ms. At worst, the number of evaluated paths will grow to 264 and it will still take less than at most 30 ms to evaluate them. Hence, the behavior planning is sufficiently fast and has room to grow computationally more complex for extensions for more intricate situations.

15.2 Aggregated Rates of Success and Misbehavior

To evaluate the overall performance, it is useful to focus on a higher level of abstraction. Figure 15.1 illustrates an exemplary 75 km stretch of highway driving. For this, the number of successful lane changes and the number of lane change abortions are counted. Figure 15.3 depicts those actions over the course of 1330 km of automated driving on the stretch from Figure 15.1.

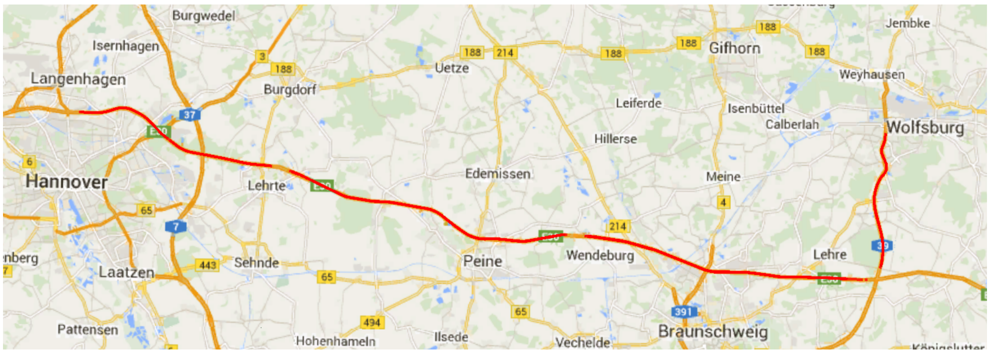


Figure 15.1: Evaluation over the course of 75 km. Source: MapQuest 2015, OpenStreetMap

Rates of Events during 1330 km

Figure 15.3 illustrates approximately 1330 km of automated driving over the course of 690 minutes. During that time, the tactical behavior planning successfully executed 566 lane changes. A lane change was aborted 46 times during its execution because the lane change situation assessment judged continuing a lane change no longer safe.

Figure 15.2 illustrates the temporal durations between indicator activations and subsequent abortions of lane changes. Lane changes were aborted in a time range between 1.25 s and 5.25 s. In total, 46 lane change abortions were recorded in 1330 kilometers of automated driving. 29 lane changes were aborted within the ego lane, 17 lane changes were aborted with a (partial) intrusion into the neighbor lane with some part of the automated vehicle.

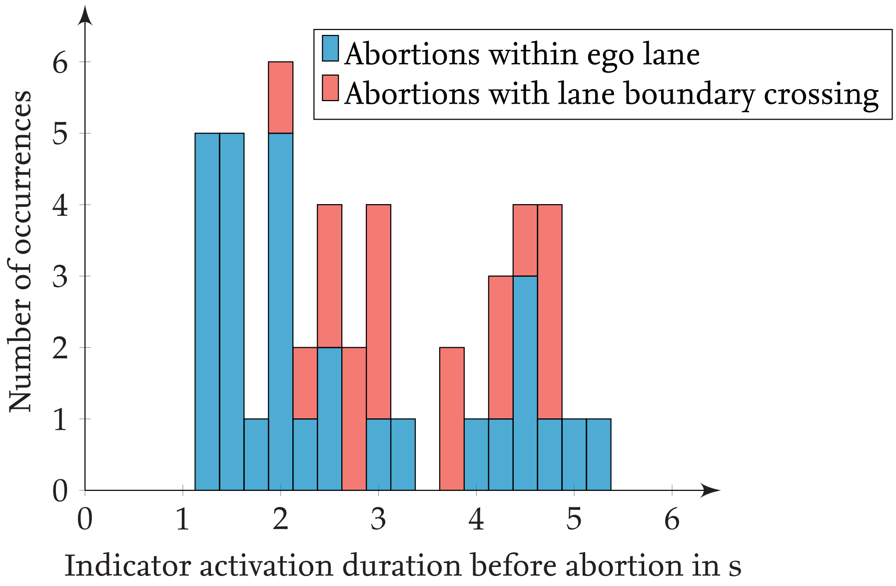


Figure 15.2: Time between indicator activations and subsequent lane change abortions

During the 1330 km ten human takeovers during lane changes were counted. Given the fact that all human drivers were driving the automated vehicle for the first time and thus were to some extent insecure about the skills of the vehicle, this is a very good result. Table 15.1 provides a detailed view of those ten takeovers.

Object Association

In the first scenario a faster vehicle from behind was perceived as being two lanes ahead. The direct neighbor lane was assumed to be narrower than in reality. Thus, the automated vehicle considered the direct neighbor lane to be unoccupied and initiated a lane change. The false association was detected and would have been corrected 1.15 s after a human driver intervened and aborted the lane change. Despite the fact that the situation might have been resolved by the vehicle itself, it would have resulted, at least, in a critical situation.

Temporary Roadworks

In the second scenario a human driver intervened in a lane change to the left because the automated vehicle did not detect that there were temporary roadworks ahead on that lane and thus the lane was blocked. It was not a malfunctioning of the lane change planning itself but rather of the traffic sign detection for temporary traffic signs.

Disliked Cooperation

In the third scenario the automated vehicle cleared the faster left lane for a pressing rear vehicle by a lane change to the right behind a slower vehicle. The human driver disapproved that slowdown, took over and steered back to the left lane. The scenario was neither critical nor a true malfunction of the system but rather a matter of taste.

Lane Marking Perception

In the fourth scenario lanes were perceived with a high angular error during a lane change. The perception error resulted in uncomfortable lateral control. The human driver took over to improve lateral control. Similarly, in the ninth scenario, a lane change was executed at a place where lane markings were missing for several meters due to prior roadworks. The human driver took over to improve lateral control. The automated vehicle did not manage to see the missing lane markings long enough ahead to stall the lane change execution in the first place.

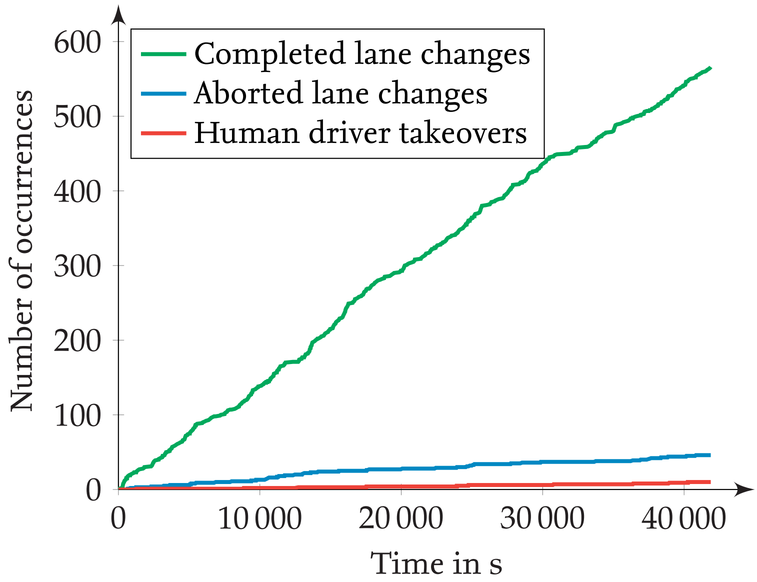


Figure 15.3: Tactical driving actions over the course of 1330 km

Not Programmed Stops

The fifth and eighth scenarios are both situations where the human driver took over during a lane change because the experiment design required the vehicle to leave the highway at the next exit. The automated vehicle intended to overtake a slower vehicle because it was not programmed to leave the highway at that exit. In the seventh scenario the human driver took over to manually drive through a highway interchange. Accidentally, he or she did that while a lane change was initiated.

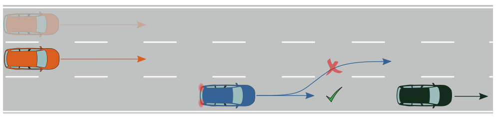
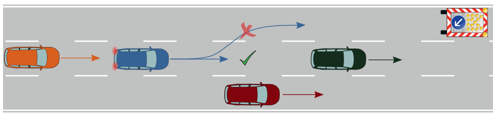
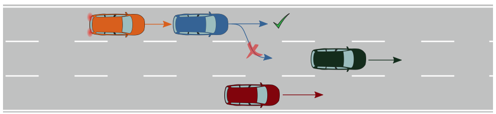
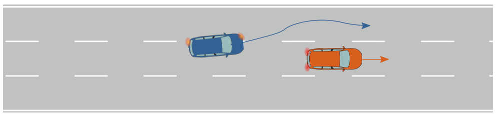
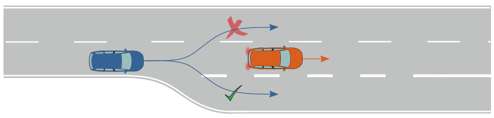
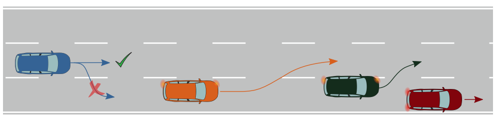
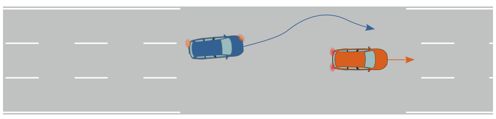
Lacking Foresight

In both the sixth and tenth scenarios the automated vehicle changed to the rightmost lane to obey the right lane driving order. A slower vehicle on the rightmost lane was hidden by faster vehicles on the rightmost lane changing to the middle lane. In scenario six it was one vehicle blocking the view, while in scenario ten two vehicles were involved in hiding the slower vehicle. A human would have been able to anticipate that a slower vehicle might be the reason for the other vehicles changing lanes and may have avoided such a lane change behind the slower vehicle.

Verdict

In total ten human driver takeovers occurred during 566 lane changes on 1330 km of automated driving. Five takeovers were caused by actual system malfunctions; three of them would have been critical without a human safety driver. The system may or may not have been able to avoid collisions in these situations. Given that there would not have been a guarantee to resolve those situations, it highlights the gap that is yet to be bridged between true, highly automated driving and still having a safety driver as a fallback level.

Table 15.1: Scenarios of human driver takeovers during lane changes

Description	Illustration	Malfunction	Critical
1) A faster vehicle from behind was associated with a wrong lane and the automated vehicle falsely assumed a lane change to be possible.		✓	✓
2) The human driver intervened in a lane change because of temporary road works on the target lane.		✓	✗
3) The human driver disapproved the automated vehicle to clear a lane for a pressing vehicle.		✗	✗
4) Lanes marking perception error occurred during a lane change; manual takeover to stabilize lateral control.		✓	✓
5, 7, 8) Manual takeover during a lane change to make the automated vehicle leave the highway at an exit where it was not programmed to exit.		✗	✗
6, 10) A lying ahead slowdown by another vehicle was hidden by one or two other vehicles.		✓	✗
9) Lane change execution while running into missing lane markings due to prior roadworks; manual takeover to improve lateral control.		✓	✓

15.3 Situation Prediction Assessment

Basing behavior decisions on a planning-ahead approach boils down to the abstract task of behavior planning into a more tangible and easier-to-evaluate task of predicting a situation correctly. Here, the behavior planning can benefit from several decades of research on prediction models.

Research Questions

Given the research efforts that have already been invested into developing situation prediction models and their level of sophistication, the author did not develop another new prediction model. The focus of this thesis is rather to provide a *framework* for behavior planning utilizing a prediction model. It is crucial to evaluate how well situations can be predicted to answer a) what prediction horizon should be considered and b) where false behavior decisions come from.

Evaluation Approach

When evaluating a situation prediction, the development of the situation in the real world may serve as a ground truth. A past situation can be predicted and compared to the recorded situation later in a sequence of recorded situations. The deviation of the predicted situation from the situation measured after the prediction time is a metric for the prediction quality.

Metric

The deviations are quantified by calculating a weighted error of object position-, velocity- and acceleration-errors, with a weighting factor of $w = 0.5$:

$$\begin{aligned}\epsilon_{obj} = &\left|s_{pos,\,obj} - s_{pos,\,ground\,truth}\right| \\ &+ w \cdot \left|s_{vel,\,obj} - s_{vel,\,ground\,truth}\right| \\ &+ w^2 \cdot \left|s_{acc,\,obj} - s_{acc,\,ground\,truth}\right|\end{aligned} \tag{15.1}$$

As a premise, it is assumed that perception errors are smaller than prediction errors. Thus, each perceived object in the measured, ground-truth situation is matched to its closest object in the same lane in the predicted situation. Overall prediction errors are summed over all objects in a situation:

$$\epsilon_{overall} = \sum \epsilon_{obj} \tag{15.2}$$

Figure 15.4 depicts the deviations from a predicted situation from a later measured, real situation as a function of the prediction horizon. This analysis only considers situations in which at least one object association for each object in the measured, ground-truth situation exists. Thus, it excludes situations in which all objects of a lane clear that lane, or all objects in a particular lane have disappeared.

Mean and Median

Figure 15.4 illustrates the mean and median of the prediction error for two different prediction models. The median error is significantly lower than the mean prediction error as it its less prone to outlier object mismatches. The error metric $\epsilon_{overall}$ is by its definition a unit-free measure. It allows a relative comparison between different prediction models.

Prediction Horizon

The prediction error increases with the prediction horizon. At a certain prediction horizon the situation deviation metric decreases its growth rate. Here, the prediction quality decreased so much that object associations become more and more

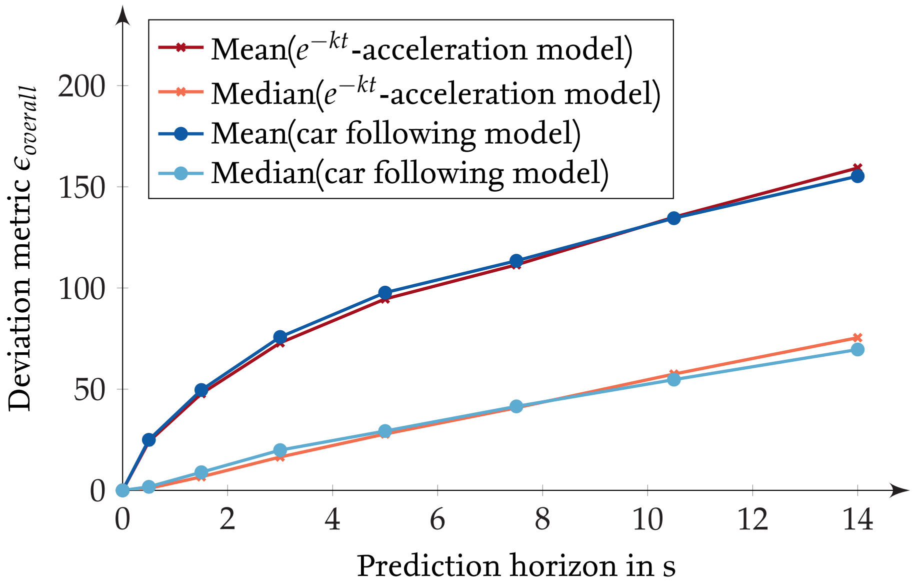


Figure 15.4: Prediction deviation metric $\epsilon_{overall}$ as a function of the time horizon for different prediction models

random. Little value could be obtained from evaluating such seemingly no longer reality related situations.

Comparison with Simple Model

Figure 15.4 not only evaluates the situation prediction model from Section 10.4 but also the exponentially decaying acceleration prediction model. Here, the acceleration is assumed to decrease over time. For small prediction horizons the given acceleration is projected into the future. For longer prediction horizons the acceleration decreases by $e^{-k \cdot t}$ with a decay time constant $k = 0.15\ \mathrm{s}^{-1}$ that essentially changes it into a constant velocity prediction model. Surprisingly, this rather simple model performs in a similar manner to complex, theoretically sound driving behavior based models if non-perfect, real measurement data is used as input.

Limitations

Analyzing specific situation predictions in detail demonstrates that the interactions between objects being modeled by the prediction model in section 10.4 do occur and are better predicted. However, the fact that current, very sophisticated prediction models provide on an overall level little gain compared to rather simple models pinpoints that further improvements in specific aspects of cooperative behavior do not address the major sources of prediction errors. These are rather owing to an insufficient handling of object existence predictions and to insufficient handling of state uncertainties.

Implications

Given that predictions are prone to errors, should a behavior decision be based on a prediction at all? From the author's point of view it should. Not only is it far easier to improve a situation prediction model than an abstract behavior decision logic, but also it is more quantitatively evaluable. Not considering the future developments of a situation seems inappropriate given that the focus of behavior decision making will gradually change towards more complex traffic situations and behavior interactions in which these future developments are central.

15.4 Safety in Performing Lane Changes on a Highway

This section provides a macroscopic analysis of lane changes in automated driving regarding the value dimensions of safety (cf. Chapter 9). The indicator activation prior to a lane change shall be in a timely manner to inform other drivers as well as the passengers in the vehicle about an intended maneuver (cf. section 15.4.1). The safe execution of a lane change for the ego vehicle itself is evaluated by analyzing the actual lateral shift towards the neighbor lane in section 15.4.2. No swinging or instability should occur here due to too dynamic lateral maneuvers. To ensure safe lane change execution within the traffic, the maximally remaining reaction time during a lane change is calculated as a safety metric (cf. section 15.4.3).

15.4.1 Indicator Activation

Legal Guidelines

Traffic regulations require the activation of an indicator before a lane change.[1] The jurisdiction does not provide clear thresholds regarding "how early" to indicate a lane change, but only qualitative statements like "in a timely manner". Freyer (2008, p. 99) analyzed the time between the indicator activation and touching the lane boundary marking with the first part of the lane changing vehicle. For that, he evaluated 615 lane changes of human drivers. Figure 15.5 presents Feyer's results (2008, p. 100) normalized by the 615 *manual* lane changes to obtain a probability density function (blue distribution). It compares this distribution with the probability distribution of indicator activation durations before an automated lane change of the test vehicle.

Human Behavior

Automated Lane Changes

The probability distribution for *automated* lane changes in Figure 15.5 is based on 550 lane changes (green distribution). On average, a lane change is indicated 3.45 s before the automated vehicle crosses the line to the neighbor lane. By the design of the algorithms a lane change is announced at least 1.0 s before the automated vehicle starts to build up a lateral displacement. According to Freyer (2008), a human driver often indicates a lane change later than that.

Specifically Different

However, it did not feel comfortable for the passengers in the car to imitate human lane changing behavior with such a small lead time for automated lane changes. Indicating a lane change early gives humans more time to understand what the car is planning, what is going on around the automated vehicle, and – if desired – the ability to override a maneuver. Even if a passenger is not required to monitor the system, it still does not feel good being unable to intervene because of an automated maneuver execution with an insufficient lead time.

Challenge

Indicating a lane change earlier makes it technically far more challenging: Objects need to be detected in higher distances and situations need to be predicted further into the future.

Indicator Activation versus Lane Changing

Yet, a false indicator activation is less severe than a false lane change with a possibly thereof resulting collision. Hence, a false indicator activation which is based on

[1] E.g. "Every lane change has to be indicated in a timely manner and clearly; for that turn signal indicator lights are to be used. (German: Jeder Fahrstreifenwechsel ist rechtzeitig und deutlich anzukündigen; dabei sind die Fahrtrichtungsanzeiger zu benutzen.)", StVO, 2013, §7 sec. 5 II.

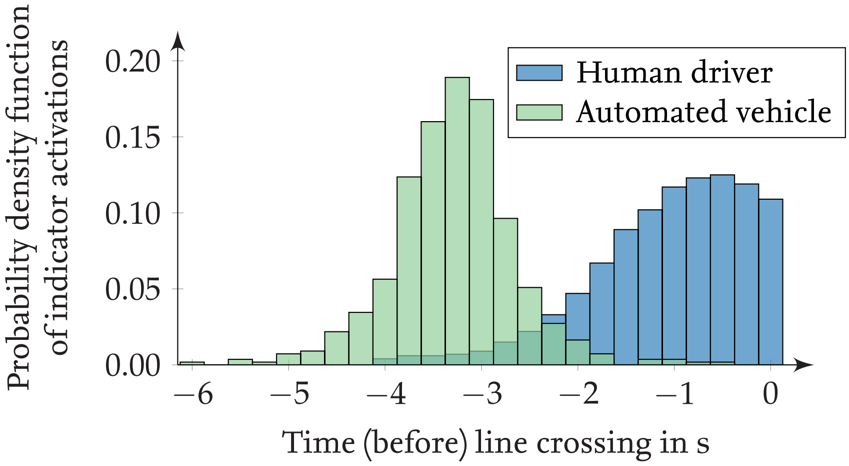


Figure 15.5: Indicator activation time before a lane change. Comparison between a human driver (based on: Freyer, 2008, p. 99) and the automated vehicle

more uncertain and possibly incorrect information, may yet be acceptable. Different rewards for indicator activation and beginning with an actual lane change allow a different parametrization to *hesitate* with the execution of a lane change within certain time limits in case of uncertain traffic situations. Section 15.2 presents time durations between indicator activations and subsequent abortions of lane changes.

15.4.2 Lateral Displacement

After a lane change has been indicated it will be executed. Therefore, a lateral displacement to the center of the ego lane will be built up. Figure 15.6 illustrates the lateral component of 261 lane change trajectories for automated lane changes. Displacement

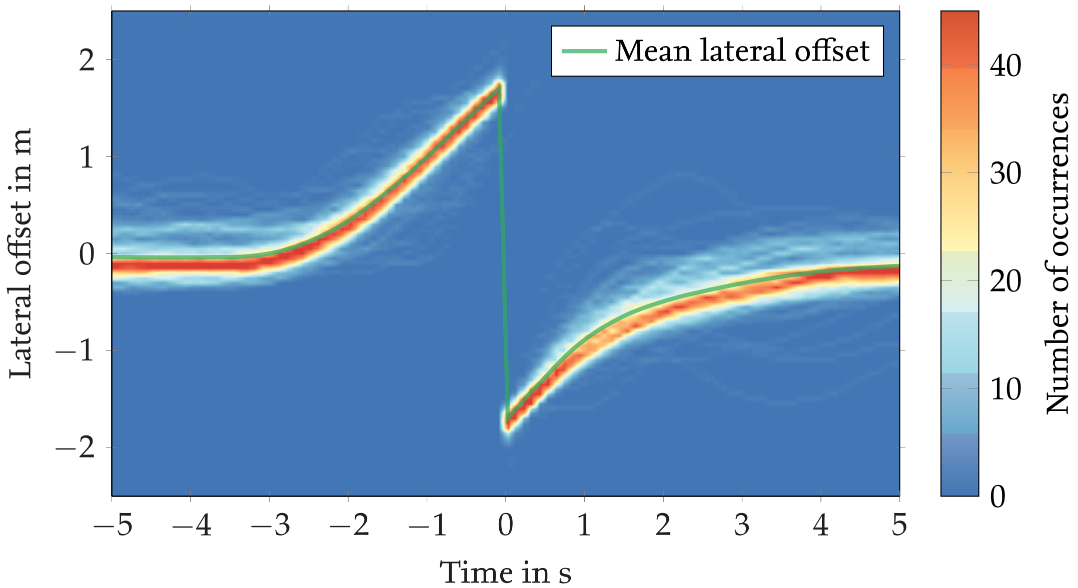


Figure 15.6: Lateral displacement of 261 trajectories for automated lane changes to the left

For the sake of readability, only lane changes to the left are depicted. Thus, a lateral displacement to the center of the ego lane starts to build up with a positive sign (left) until the lane marking is crossed and the reference ego lane jumps to the former left neighbor lane. After such a jump, the lane change is completed by re-centering the automated vehicle to the new lane.

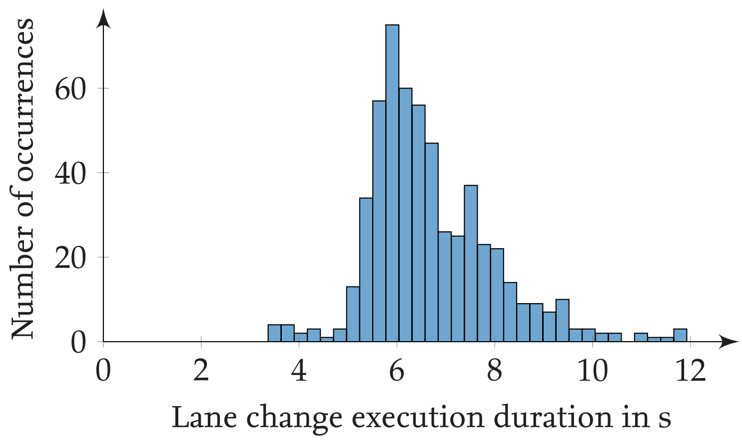


Figure 15.7: Histogram of completion times for a sample of 561 automated lane changes

Completion Time

Figure 15.7 shows a histogram for the completion time of 561 automated lane changes; both to the left and right. On average a lane change completion takes 6.8 s.

15.4.3 Maximally Available Reaction Time during Lane Changes

Maximally Available Reaction Time

Section 12.2 suggested to use the maximally available reaction time $t_{react,\,max}$ to quantify safety in respect to other vehicles (cf. Chapter 9). The results are compared to those of a human driver in Habenicht (2012, p. 91 ff.). $t_{react,\,max}$ is calculated by:

$$t_{react,\,max} = \frac{\left|s_{pos,\,behind}\right| - (s_{vel,\,other} - s_{vel,\,ego})^2/(2 \cdot b_{max})}{s_{vel,\,behind}} \tag{15.3}$$

$\left|s_{pos,\,behind}\right|$ is the positive distance between another vehicle and the ego vehicle. $s_{vel,\,other} - s_{vel,\,ego}$ is the relative velocity difference between the ego vehicle and a second vehicle. $s_{vel,\,behind}$ is the velocity of the behind vehicle (cf. attachment D). b_{max} is the maximal deceleration to avoid a collision. $b_{max} = 10\,\mathrm{m/s^2}$ is defined to make the evaluation comparable with Habenicht (2012, p. 91 ff.). Similar as in Habenicht, the maximally available reaction time is calculated relative to three vehicles (cf. Figure 10.15): Towards the front vehicle in the ego lane (FE), towards the front vehicle in neighbor lane (FN), and towards the rear vehicle in the neighbor lane (RN).

Comparability of Results

Slightly different criteria were used for the evaluation time. Habenicht (2012, p. 93) used the steering angle as a criteria to determine an evaluation period. Here a time window of 6 s centered around the time of line crossing is used. In accordance with Habenicht (2012, p. 94), a two-sided Wilcoxon rank-sum test is used with the null hypothesis that the distributions of the $t_{react,\,max}$ of a human driver and the automated vehicle are equal. Habenicht (2012) does not explicitly state if he only considers lane changes to the left in his curves or also lane changes to the right. Based on the way the curves look like and because the scenario is described by the

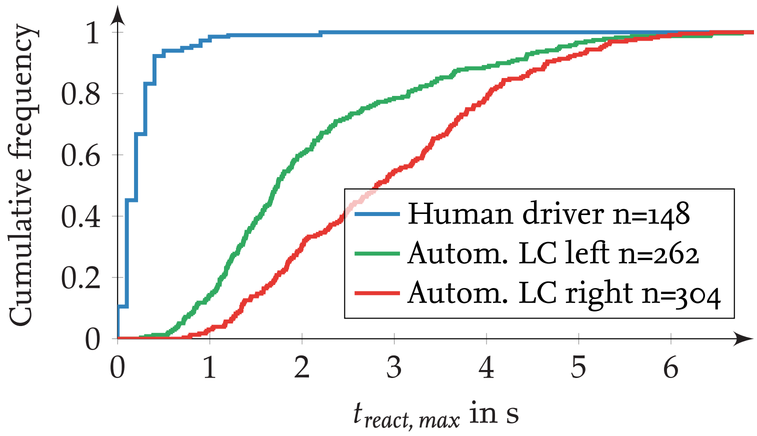


Figure 15.8: Cumulative frequency of $t_{react,\, max}$ during lane changes (LC) towards front neighbor vehicle. The data for human drivers is based on Habenicht (2012, p. 95)

example of a left lane change (Habenicht, 2012, p. 94), the author assumes only lane changes to the left were evaluated by him.

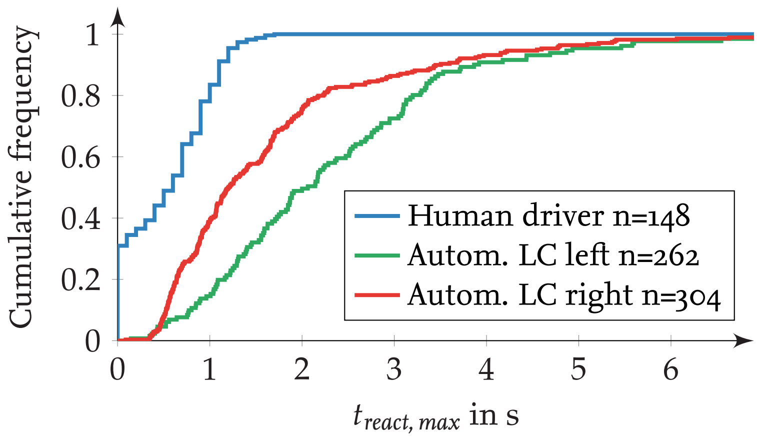


Figure 15.9: Cumulative frequency of $t_{react,\, max}$ during lane changes (LC) from rear neighbor vehicle towards the ego vehicle. The data for human drivers is based on Habenicht (2012, p. 95)

Cumulative Frequency

The Figures 15.8, 15.9, and 15.10 show the cumulative frequencies of the maximally available reaction times during lane changes regarding the front neighbor vehicle, of the rear neighbor vehicle towards the automated vehicle, and of the automated vehicle towards the front ego vehicle.[2] The data from 566 automated lane changes is compared with those from 148 lane changes of human drivers in Habenicht (2012). It is separated into 262 automated lane changes to the left and 304 automated lane changes to the right. The further right a curve is in the plot, the higher is the maximally available reaction time. All three evaluations show a statistically significant difference between the maximally available reaction time of the automated vehicle during lane changes over that of human drivers. Yet, for a small fraction of the lane changes the maximally available reaction time is below the approximately one second of reaction time a human driver would need. A dedicated analysis of these

[2]Nomenclature refers the vehicles' roles *before* executing the lane change.

scenarios showed that these were typically due to false object detections and thus a lack of true ground truth data for evaluation.

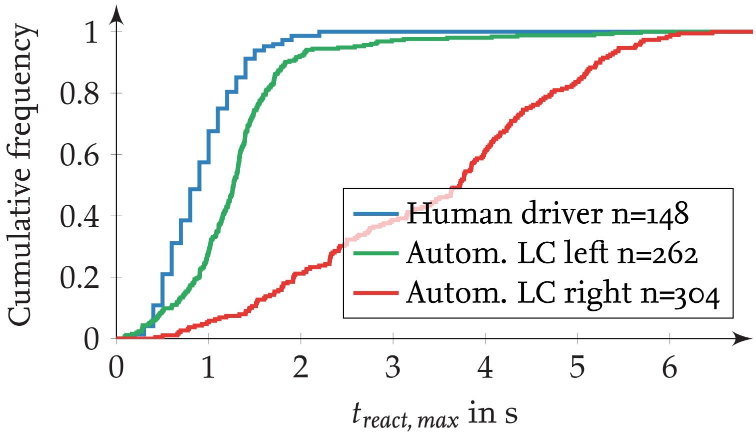


Figure 15.10: Cumulative frequency of $t_{react,\,max}$ during lane changes (LC) towards front ego vehicle. The data for human drivers is based on Habenicht (2012, p. 96)

Towards Front Vehicles

The maximally available reaction time towards a front vehicle in the ego or neighbor lane is more critical for lane changes to the left than it is for lane changes to the right. The experiment was conducted on a German highway where overtaking is only allowed on the left. Hence, velocities on the left lane tend to be higher. Most lane changes to the left are motivated by a slow front vehicle in the ego lane. The maximally available reaction time is most of the time between 0.9 s to 2 s. This reflects a lane change out of a front vehicle following or approaching situation. Lane changes to the right often occur with far higher maximally available reaction times. These lane changes may be due to clearing a lane in accordance with the right lane driving order (cf. section 10.3.1), reaching a navigation destination, or simply clearing a lane for a tailgating rear vehicle (cf. section 10.5.1).

Towards Rear Vehicles

The maximally available reaction time of a neighbor rear vehicle towards the automated vehicle during an automated lane change is also higher than that of a human driver for both, lane changes to the left and right. Interestingly, it is smaller for lane changes to the right than it is for lane changes to the left. Analyzing the scenarios, this is due to frequently occurring cut-ins in front of slower vehicles on the right lane after overtaking them with higher velocities. For lane changes to the left, the automated vehicle is frequently faced with faster vehicles from behind. These need to be reliably detected and tracked. If there is no speed limit, these vehicles may be significantly faster than the ego vehicle (cf. section 14.2 and the counter ability restriction in section 10.3.2). Under consideration of the uncertainty in these scenarios it seems legit to make automated lane changes more conservative than those a human driver would risk.

Limitations regarding Safety

All in all, this evaluation illustrates that the value dimension of safety has well been considered in the development of the automated lane changing feature. However, what the statistical evaluation in this section cannot provide is to describe safety in those rare cases where the scenario deviates from regular driving. Yet, those outliers are relevant because they are at high risk to result in a crash. Hence, there is a dedicated analysis of those few critical situations that occurred during the total evaluation distance of 1330 km in section 15.2.

15.5 Mobility: Lane Change Velocity Gain

Mobility

Section 12.2 highlighted the challenges of providing a metric for the value dimension of mobility (cf. Chapter 9). Mobility entails abstract aspects like availability of the automated driving system in different domains and situations. Moreover, it entails mission fulfillment. Certain missions may not be executable if the automated vehicle cannot pass a highway interchange (cf. section 14.3). If a mission can be fulfilled, metrics like the total travel time will be of relevance. Yet, such a metric is susceptible to several external factors (cf. section 12.3).

Velocity Gain

Hence, this evaluation focuses on the velocity gain from performing single lane changes. Here, the velocity of vehicles on the neighbor lane is compared with that of the ego vehicle on the ego lane prior to the lane change. Figure 15.11 illustrates the average velocity difference. The average velocity gain is 3.24 m/s for 262 lane changes. The median is 2.48 m/s. The evaluation was conducted on a German highway where overtaking is only allowed on the left. Hence, only lane changes to the left were considered.

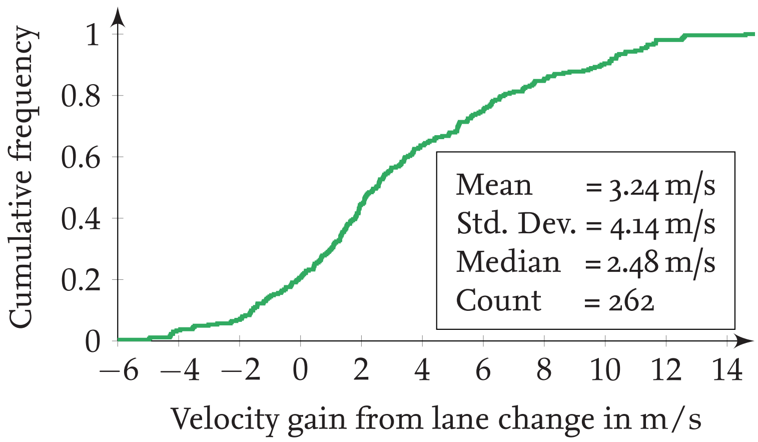


Figure 15.11: Cumulative frequency lane change velocity gain

Negative Velocity Differences

20.6 % of the lane changes where performed although the velocities of the vehicles on the left lane was slower. While a certain amount of those instances in the evaluation is rather due to noise in the velocity estimation of those neighbor vehicles, there are likewise situations where there is indeed reason to change to the left lane. As described in section 10.3.1, if the current ego velocity is lower than the target velocity the automated vehicle will still try to get to the leftmost lane assuming that once traffic speeds up, it can only here overtake other vehicles. This is due to the general rule that no overtaking on the right is allowed on highways[3]

[3] If velocities are below a certain threshold as in a traffic jam, it is likewise allowed to overtake on the right. Here the automated vehicle will also overtake on the right.

15.6 Subjective Assessment by Others [4]

In section 12.1 different methods for performance evaluation were presented. In Ulbrich & Maurer (2014) the authors presented an approach for assessing the lane change planning performance against a ground truth. Likewise, the author tried to obtain a ground truth whether a lane change is possible or beneficial for highway scenarios. Unfortunately, human drivers judged the situation very inconsistently. Therefore, such a ground truth evaluation provides little value. Obtaining quantifiable ground truth references for behavior planning is yet to be solved by follow-up research efforts. This thesis will limit its evaluation to the earlier presented quantitative analysis of single maneuvers, quantitative analysis of physical quantities like durations or distances and a subjective assessment by human drivers/passengers.

A subjective assessment, e.g., by a group of test persons, permits a performance evaluation on a high abstraction level. Section 15.6.1 presents the experiment design for this study. The section is completed by evaluating the subjective assessment results.

15.6.1 Experiment Design and Conduct

Table 15.2 presents the properties of the pool of test persons who were consulted for a subjective assessment of the lane change function. In total, 5300 kilometers were driven by 28 test persons for this study. 71% of the test persons were male and 29% were female. Their age span was 23 to 56 years old with 5 to 39 years of driving experience, respectively. Similarly, driving experience varies between those test persons in a wide range from 5000 km per year to 100 000 km per year. The test persons characterized their driving style in the range from "defensive" to "very sporty". To reduce the impact of inexperience on automated driving, the focus of this study was on test persons with prior experience with driver assistance systems.

15.6.2 Subjective Assessment by the Test Persons

The test persons were asked to complete the questionnaire in attachment H. Table 15.3 presents a detailed assessment of current challenges in lane change planning.

Lane Change Motivation

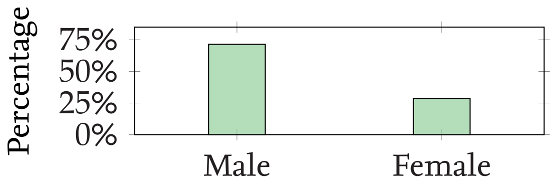
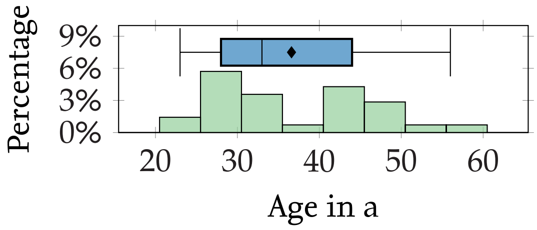
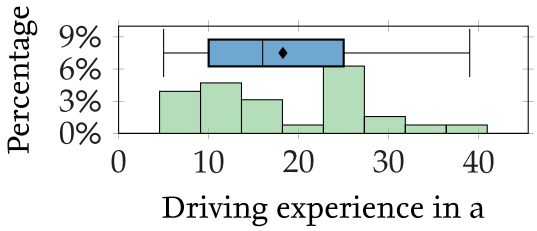
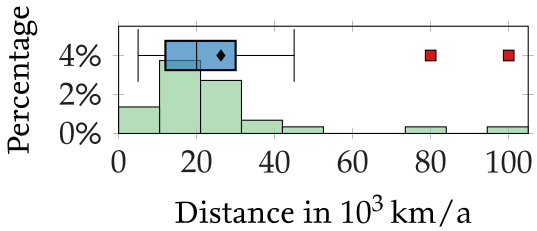
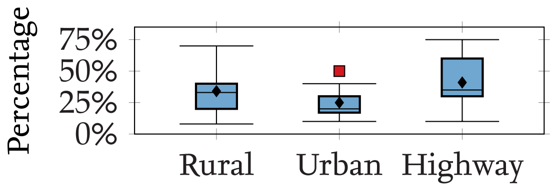
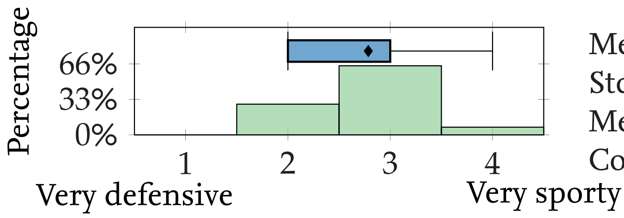
The majority of the test persons would have preferred it if the automated vehicle performed lane changes due to smaller individual disadvantages (R1). At the same time, the majority judges the behavior still appropriate for an automated vehicle (R2).

Defensiveness vs. Sportiness

Likewise, a clear majority judges the lane change behavior to be rather "defensive" or "very defensive" in a range between "very defensive" to "very sporty" (R5). A wide

[4] My colleagues Amelie Stephan and Ina Othersen served as experiment supervisors during the test person study, managed the experiment execution, and contributed to the design of the questionnaire for lane change relevant items. Their primary focus for this study was on human-machine interface design. The lane change evaluation was a byproduct of this study. The author contributed the test items for the evaluation of lane change relevant aspects and to the technical experiment conduct as well as the acquisition of relevant data for the study.

Table 15.2: *Characteristics* of the 28 test persons

Attribute	Illustration	Statistics
C1: Gender		Mean=(1.29) Std. Dev.=(0.46) Median=(1) Count=28
C2: Age		Mean=36.61 Std. Dev.=9.37 Median=33.5 Count=28
C3: Years of driving experience		Mean=18.25 Std. Dev.=9.39 Median=16 Count=28
C4: Driving experience per year		Mean=26 232 Std. Dev.=20 527 Median=20 000 Count=28
C5: Share of driving experience in different domains		Mean=34/25/41 Std. Dev.=18/10/19 Median=33/23/35 Count=28
C6: Driving style		Mean=2.79 Std. Dev.=0.57 Median=3 Count=28

range of answers was given to whether this defensiveness/sportiness trade-off is appropriate for an automated vehicle (R6). Most test persons disagreed, but no test person expressed “strong dissent”. Slightly less than 20% of the test persons expressed a “strong approval”, “approval”, or “neutral” response.

Link to Technical Limitations

Hence, there seems to be a preference towards rather defensive driving behavior for an automated vehicle, but at the same time test persons would prefer a bit less defensive driving. This addresses a current technical limitation of the tactical behavior planning: To reduce the number of maneuver abortions and exposure to critical driving situations, priority has intentionally been given to “safety” over “user satisfaction” (cf. Chapter 9). This comes at the cost of accepting bigger individual disadvantages for the automated vehicle (R1) and less sporty driving behavior.

Gap Selection

The gap selection was rated quite inconsistently (R3). Most likely, the results were strongly influenced by single situations where the automated vehicle either successfully or not successfully targeted specific gaps while each test person was in the driver’s seat. Thus, it does not seem possible to derive an aggregated assessment without looking specifically into each gap adjustment situation. Moreover, certain improvements were made to the gap adjustment which were not part of the software release during the test person study.

Lane Change Execution

After a gap was targeted, the lane change execution itself was assessed rather positively. 60% of the test persons responded by giving their “approval” or “strong approval” that the lane change execution is appropriate for an automated vehicle. Only 29% voted for “dissent”; no one expressed “strong dissent”.

Commanding Driving Maneuvers

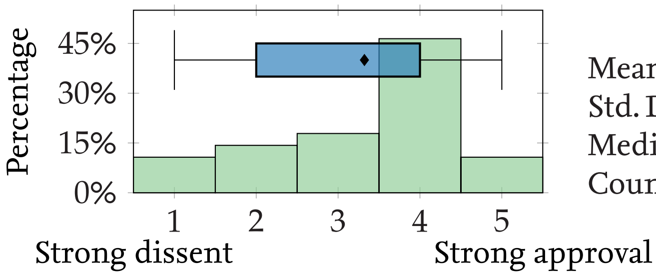
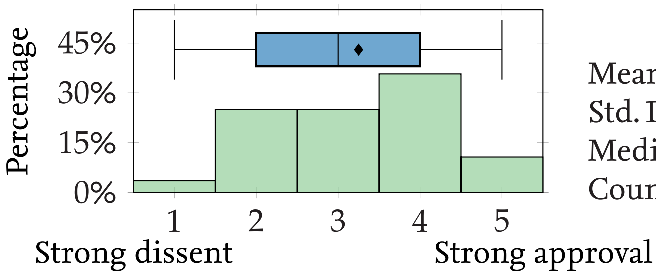
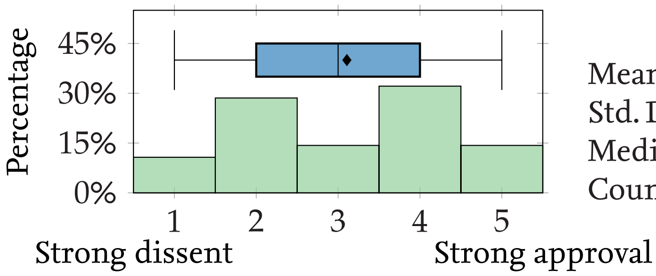
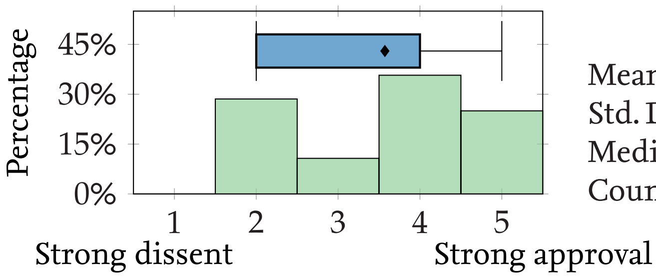
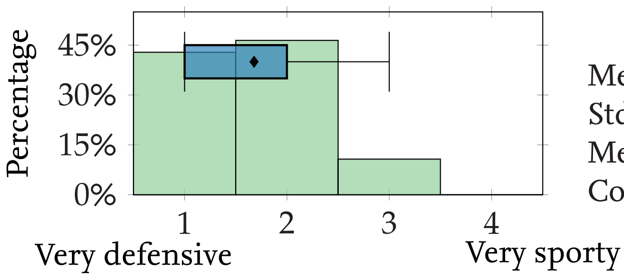
The majority of the test persons would prefer to have an (additional) *option* of commanding tactical driving decisions (R7). At first sight, this clearly contradicts the concept of highly automated driving. Yet, providing *additional, optional* input to command tactical maneuvers may alleviate the current technical limitations of the automated vehicle and still provide a better overall experience. Given a driver *is* back in the loop of the driving task that he or she is able to dislike a certain driving behavior, he or she may very well have the *option* of changing it within the automated driving mode instead of resuming entirely manual driving.

Aborting Maneuvers

Figure 15.3 illustrates that approximately 8% of the lane changes are currently not successfully executed and are aborted during the maneuver. Despite the lack of quantitative data on how many lane changes are aborted by human drivers in similar traffic, this number seems higher than the typical rate of lane change abortions of human drivers.[5] However, only a minority of 20% of the test persons voiced their “approval” that lane change abortions were disturbing. No one expressed “strong approval” that lane change abortions were disturbing. A majority did not find lane change abortions disturbing. Hence, it seems that there is a certain tolerance for overthrowing tactical decisions and thus aborting maneuvers should certainly be part of any tactical behavior planning.

[5] Wakasugi (2005) performs a driving study and reports 266 human lane change abortions while analyzing 1097 human executed lane changes on 1500 km on a Japanese expressway. Yet, the transferability of the results seem questionable to the here presented test setup and environment. To the author’s non-quantitative judgment, the automated system performs not as well as a human driver would do.

Table 15.3: *Results* of the lane change behavior assessment by 28 test persons

Attribute	Illustration	Statistics
R1: Preference for automated vehicle to perform lane changes due to smaller individual disadvantages		Mean=3.32 Std. Dev.=1.19 Median=4 Count=28
R2: The motivation to execute lane changes is appropriate for an automated vehicle		Mean=3.25 Std. Dev.=1.08 Median=3 Count=28
R3: The selection of eligible gaps for lane changes is appropriate for an automated vehicle		Mean=3.11 Std. Dev.=1.29 Median=3 Count=28
R4: The execution of lane changes is appropriate for an automated vehicle		Mean=3.75 Std. Dev.=0.93 Median=4 Count=28
R5: Assessment of the lane change behavior of the automated vehicle in its defensiveness/sportiness		Mean=1.68 Std. Dev.=0.67 Median=2 Count=28

Continued on next page

Table 15.3: - continued from previous page

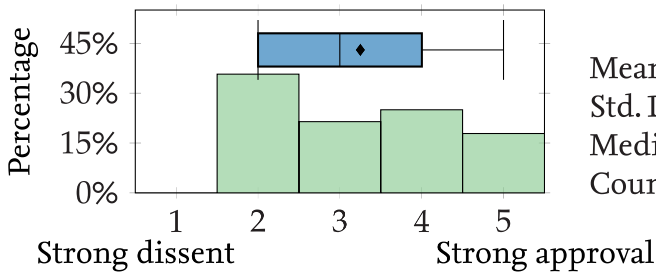
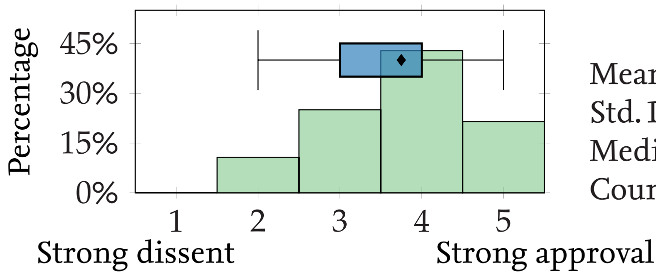
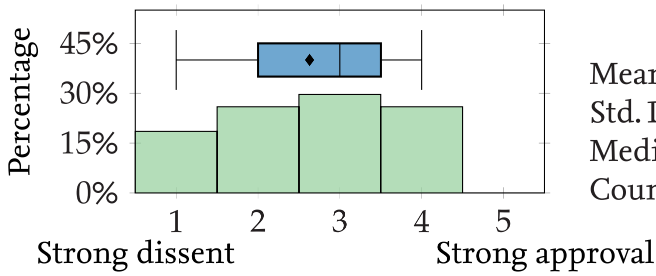
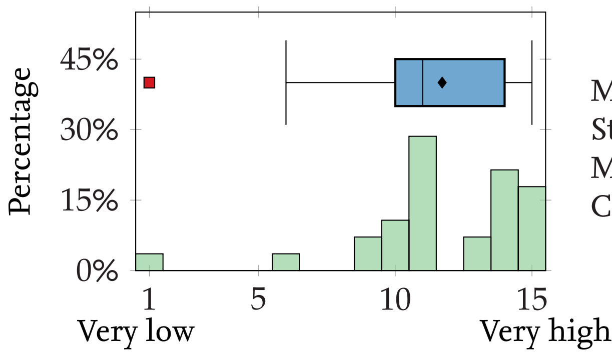
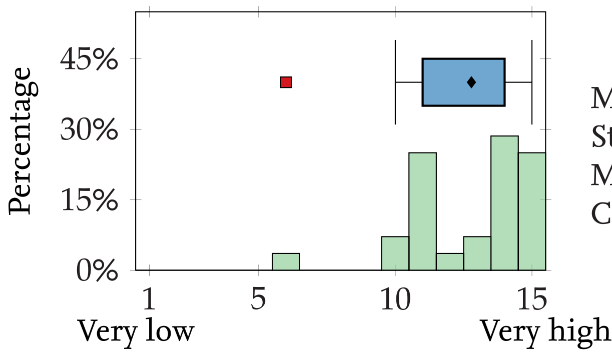
Attribute	Illustration	Statistics
R6: The lane change behavior in its defensiveness/sportiness trade-off is appropriate for an automated vehicle		Mean=3.25 Std. Dev.=1.14 Median=3 Count=28
R7: Having the *option* of commanding tactical driving decisions to the automated vehicle would be preferred		Mean=3.75 Std. Dev.=0.93 Median=4 Count=28
R8: Abortions of already initiated lane changes were experienced as disturbing		Mean=2.63 Std. Dev.=1.08 Median=3 Count=27

Table 15.4 presents an overall assessment of the automated vehicle as a whole and the lane change behavior planning in particular. Both overall assessments are skewed towards positive assessments. A grading scale from "very low" (1, 2, 3), "low" (4, 5, 6), "neutral" (7, 8, 9) to "high" (10, 11, 12), and "very high" (13, 14, 15) has been used. A strong majority judged the overall system performance and the lane change performance as "high" or "very high".

Statistical Significance

With a significance level of $1-\alpha = 95\%$ and a working hypothesis H_1 that the overall assessment OA2 (cf. Table 15.4) is higher than that of OA1 yields the null hypothesis H_0 that the assessment in the overall assessment OA1 is the same or less than OA2: $H_0 : \mu_{OA1} - \mu_{OA2} \leq 0$. It is a single-tailed test of paired samples. Hence, a student-t test can be applied with a standard error of the sampling distribution of: $SE = \sqrt{\sigma^2_{OA1-OA2}/n} = \sqrt{2.801^2/28} = 0.529$, resulting in a $t = \mu_{OA1-OA2}/SE = -1.071/0.529 = -2.024$. For a student t distribution with $n-1 = 27$ degrees of freedom, this results in a probability of $p = 2.64\%$ that the null hypothesis is true being slightly higher than $\alpha/2 = 2.5\%$ for a single-tailed hypothesis test.

Table 15.4: Comparing the *overall lane change assessment* with an *overall automated driving assessment*

Attribute	Illustration	Statistics
OA1: Test persons' assessment of the driving performance of the automated vehicle during lane changes		Mean=11.71 Std. Dev.=3.13 Median=11 Count=28
OA2: Test persons' assessment of the driving performance of the automated vehicle as a whole		Mean=12.79 Std. Dev.=2.20 Median=14 Count=28

Thus, the overall assessment is not – in a statistical sense – significantly better, but yet it is descriptively better than the assessment for lane changes. To the author, this is not a surprising finding: Several factors contribute to the overall assessment that are far less at the edge of technical feasibility, for instance, longitudinal distance keeping, jerk minimal driving, or lateral lane following. Most of these are far more state of the art than lane change behavior planning and require – in comparison – less sophisticated perception skills.

Interpretation of Results

All in all, the assessment illustrates a high level of satisfaction with the system performance among the test persons. Only two test persons gave a "low" or "very low" grading of the lane change assessment. Why they gave such a low rating cannot fully be explained by the data. Neither of the two test persons experienced a dedicated malfunctioning of the system during lane changes. Only the test person giving a "very low=1" rating of the automated vehicle during lane changes experienced two manual takeovers due to two highway stretches with missing road markings caused by prior road construction works. Surprisingly, he rated the lane change performance as "very low=1", while rating the overall performance as "high". The other test person giving a "low" rating did not experience any system failures.

Outliers

The pie chart in Figure 15.12 shows that a strong majority of 89.3% of the test persons would like to have such an automated driving system with the capability

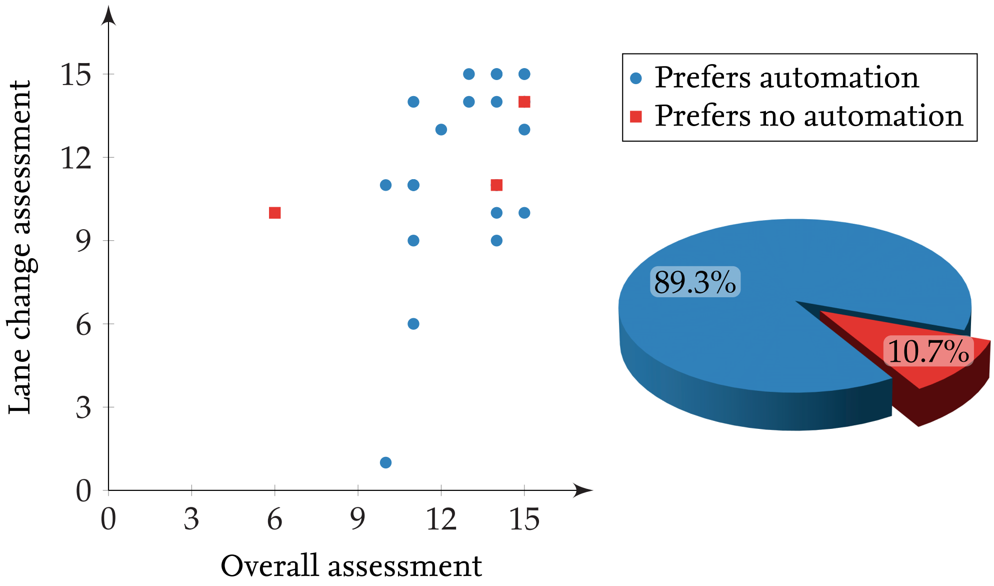


Figure 15.12: Scatterplot of lane change and overall system grading of test persons who would like or would not like to have the demonstrated automated driving system in their own vehicle

For Test Persons' own Vehicles?

of highly automated driving in their own vehicle.[6] The scatter plot in Figure 15.12 illustrates that, surprisingly, the willingness to drive such an automated vehicle is not clearly influenced by the test persons' grading of the overall system performance or the performance grading for lane changes. One should note that the rate of people who would like to have such a highly automated system may be biased by the study design itself: To reduce the impact of inexperience with automated driving, the focus of this study was on test persons with prior experience with driver assistance systems. Thus, the test persons may be more tech-savvy and inclined towards automation than regular drivers.

15.7 Conclusions

In this macroscopic evaluation the focus was not on single maneuvers but rather on aggregated statistics over more than 1300 km of automated driving in real traffic. In the requirements section, it was stated that behavior planning for lane changes should be fast, consistent, provident, deterministic, and in compliance with the system's values.

Sufficiently Fast

The evaluation of the computational complexity showed that the algorithm is sufficiently fast to be executed in real-time on any standard computer. Thus, it leaves adequate computational resources for considering additional aspects.

Completions and Abortions

Consistency in its behavior is assessed by analyzing lane change completions and abortions. Evaluating an aggregated rate of successful lane change maneuvers com-

[6] In this study, "highly automated driving" was coined as the mode where all longitudinal, lateral, and tactical behavior planning was performed by the vehicle. The test person was not required to drive with hands-close to the steering-wheel or monitor the system. However, a safety driver on the co-driver's seat supervised the system making it technically "partially automated driving."

pletions and lane change abortions showed that approximately 8% of the initiated lane change maneuvers are aborted. Human (safety) drivers intervened in only ten out of 566 lane change scenarios. Five scenarios were actual malfunctions of the system. Three of them would have been critical without human intervention.

Prediction

The situation prediction is a core component for provident behavior planning. Predicting driving behavior ahead is quantitatively evaluated against the true future development of a recorded situation. The prediction error has been quantified as a function of the prediction horizon.

Deterministic

The algorithms are by design deterministic. No black box models like neuronal networks are used. Thus, no evaluation of the algorithm's determinism is necessary.

Compliance with Values

Last of all, behavior planning shall be in compliance with the automated vehicle's system of values. According to Chapter 9, relevant value dimensions are safety, mobility, legality, user satisfaction, and third party satisfaction.

Safety and Mobility

For safety, the indicator activation is evaluated regarding its early timing. The automated vehicle indicates lane changes earlier than a human driver. Lane change execution is evaluated regarding its safe execution by analyzing the lateral offsets during lane changes itself and analyzing the maximally available reaction time during lane changes. Towards all surrounding vehicles, the automated vehicle maintains a higher maximally available reaction time than human drivers do. The analysis of the velocity gain from a lane change is used as a metric to quantify at least some aspects of the value dimension of mobility. It shows that there is a velocity gain from the ability of executing lane changes.

Legality

The value dimension of legality is not specifically evaluated because there is no intentional non-compliance with laws implemented in the automated vehicle. Uncertainty in perception, prediction, and behavior execution may still result in unintended behavior not compliant with rules given by the road authorities. In the worst case this may not only be crossing a solid line but even a crash with product liability implications. Yet, it can hardly be evaluated because it is no active tradeoff the automated vehicle made under the assumption of violating a law.

User Satisfaction

A subjective assessment was conducted to quantify user satisfaction. The majority of the test persons judged the lane change planning as well as the overall automated vehicle positively. Surprisingly, their assessment seems uncorrelated to whether they would like to have such a system in their own vehicle. Many drivers would at least like to have the option of commanding tactical maneuvers despite higher degrees of automation. For the author of this thesis, it was surprising that despite the rate of maneuver abortions (8% is surely higher than a human driver's rate), it did not seem too disturbing for the test persons in the automated vehicle. This is a remarkable result given that aborting a maneuver is a technical necessity for tactical behavior planning in real world driving with an imperfect perception system. The fallback option of a maneuver abortion may be the most significant step towards a robust solution working not only in selected scenarios but also in most traffic conditions.

Aborting Maneuvers

The value dimension of third party satisfaction has not been evaluated in this thesis. While a subjective assessment with test persons in the automated vehicle was challenging already, performing the same for third parties outside the vehicle could not be conducted in the scope of this thesis.

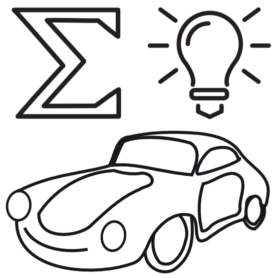
Limitations

What did this evaluation not provide? The evaluation would be a lot stronger if it could compare different implementations of tactical lane change behavior planning. Unfortunately, many of the lane change implementations used by leading teams are not open source. Often, not even the concepts have been published but only the results have been demonstrated in short video clips. Even if their implementations were available they would be highly tailored towards their specific vehicle platforms and perception systems. Thus, comparing them on the basis of the same sensor data may still not be meaningful. Only testing full vehicles in specific, reproducible scenarios would result in representative benchmarking. However, different implementations may still pursue different behavior options in similar scenarios. More than one behavior option may be feasible, thus it is still difficult to quantitatively assess the behavior planning itself.

Test Persons

The qualitative assessment based on the responses from test persons may also be biased. On the one hand, there is a self-selection of people participating in such studies despite trying to cover a representative sample of test persons as illustrated in section 15.6.1. On the other hand, training and experience with automated driving may strongly influence the test persons' responses. All test persons had prior experience with driver assistance systems. Yet, this is most likely the first automated vehicle any of the test persons have driven. Thus, they may lack reference points for performance comparison.

16 Conclusions and Outlook



All in all, what did this thesis contribute to research on automated vehicles? It provided a systemic view to entail the core aspects and implications of tactical behavior planning. Specifically, it presented the fundamentals for lane change behavior planning, offered a concept for lane change behavior planning and its implementation, and evaluated the performance of that approach.

Fundamentals

In the fundamentals, this thesis presented a consistent terminology and drew the larger picture by locating and embedding tactical behavior planning in a functional system architecture. Cooperation aspects were structured by two main dimensions and a literature review was provided. First, the literature review focused on a general framework for decision making and methods for probabilistic planning with dynamic Bayesian networks, model predictive control, and Markov decision processes. Based on this, application specific concepts and implementations for lane change related aspects were reviewed. Focus was placed on lane change situation assessment, driving behavior and situation prediction, and overall behavior planning.

Concepts and Implementations

The *concepts and implementations* part of this thesis defined requirements, provided a concept and implementation for relevant context modeling for lane change planning, related behavior planning to an underlying system of values, and described the tactical lane change behavior planning implementation itself. The implementation has been split into three main models and one behavior planning core. This entailed a measurement model to translate perceived information into a best possible state estimate, a situation prediction model for predicting a situation for a certain time in the future, and a reward and cost model to assign rewards for executing specific actions in specific situations.

Metrics and Evaluations

The developed implementation has been evaluated in the third part of this thesis. It provided a review of metrics for performance evaluation. Based on this, a simulation-based performance evaluation has been presented. It is a first necessary criterion for performance evaluation but only real world driving is a sufficient criteria. For real world driving, single maneuvers were analyzed on a microscopic level. Yet, in real world driving, a passenger in the automated vehicle will rather have a macroscopic perspective: Here the focus is not on single maneuvers but rather on the overall performance. After evaluating different phases in the lane change planning, a test person study was presented for a subjective assessment of the lane change behavior as a whole.

16.1 Current Limitations and Outlook

In this thesis a viable concept and implementation for lane change behavior planning has been presented. It has been demonstrated on several occasions to the public and tested and tweaked for more than 50 000 km in real world traffic. Yet, it is not perfect. It is likely that the Pareto principle also holds true for lane change planning: 80% of the effects may have been solved with the first 20% of the efforts. Vice versa, this means that it will still take five times the effort to fix the remaining issues accordingly.

Traffic Density & Cooperation

One key driver for the overall lane change performance is the overall traffic density. The denser the traffic is the more complicated lane changes will become. Thus, lane changes will require more cooperative interaction and are further limited by false classifications and state estimates. Improving cooperative behavior has been addressed in this thesis. However, it seems that only the tip of the iceberg has been addressed so far. There is still decades of research to be done until really human-like driving is fully achieved. Similarly, intention recognition or detecting intended or even unintended gestures is still in its infancy and is a field for future research.

Perception

Other key drivers are perceptual limitations. Lane change behavior planning requires a high level of perception skills; far more than adaptive cruise control driving or almost any other driving function. The better a future perception system is, the smaller the uncertainties will be for perceived distances, velocities, and accelerations as well as the inconsistencies in object detection. Smaller uncertainties and better consistency will translate into less defensive maneuvers and more consistent lane change behavior.

360° Object Perception

Points of improvement for the perception system of automated vehicles are manifold and beyond the scope of this thesis. Yet, one major point of improvement is to extend the sensor setup as well as object tracking to allow true 360° object perception. So far, the sides of the automated vehicle "Jack" can only be covered with less than optimally performing radar systems covered by the plastic material of the bumpers (cf. Chapter 14 and attachment A.5). A 360° multilayer laser scanner like a Velodyne HDL-64 would help to accommodate this issue but may not be acceptable from a design point of view. The next step for the team will be to find and integrate a sensor setup that allows true 360° perception and can still be nicely integrated into the vehicle design and meet target system costs. Another issue related to lane change planning is the sensor viewing range of current lane perception camera systems. Detecting the course of a lane may work maybe 60 m in front of the vehicle at most. Detecting lane marking types will be even less foresighted. Likewise, reliably detecting if there is a second or third neighbor lane is still very challenging for today's camera systems.

Lane Perception

Situation Prediction

Currently, the situation prediction is among the main points for further improvement. According to the evaluation in section 15.3, the current, sophisticated prediction model is hardly any better than a simple, exponentially decaying acceleration model. No extensive efforts have been dedicated to specifically improve the prediction model. A simple analysis of what causes the prediction to go wrong in specific situations may radically improve the prediction performance. For instance, accelerations are currently not limited from above or below. Thus, the prediction model may currently assume an acceleration of more than $10\,\mathrm{m/s^2}$. Improving the situation prediction will directly translate into a more foresighted driving behavior.

Defensive-ness

Overall, the lane change behavior is more defensive than how a human driver would drive. To a certain extent this may be justifiable for an automated vehicle rather chauffeuring passengers than proactively driving like a sporty driver. Yet, even for an automated vehicle a less defensive driving style than the current one seems desirable. However, less defensive driving goes along with more challenging driving situations, more inconsistent behavior, and possibly more support needed from a safety driver. Thus, less defensive driving will rather be a parameter to translate higher system capabilities into a better driving experience.

Urban Domains

The focus for this thesis was mainly lane change behavior planning on highways. Likewise, automated driving including lane changes has been demonstrated on rural roads. Automated lane changes have also been tested in urban domains. Given that lane changes require two lanes and no lane changes in oncoming traffic have been implemented so far, even roads in urban domains are from the aspect of lane changes not radically different to highways with traffic jams or slow moving traffic. Yet, as mentioned before, traffic densities are a key performance factor. They tend to be higher and more cooperative interaction may be required for lane change planning in urban domains. Thus, the automated vehicle will currently require bigger gaps than a human driver will need in a city. Automated driving in urban domains has not yet been demonstrated to the public by the *Audi A7 piloted driving concept* vehicle. Currently, lane changes do not look like the most striking issue towards achieving this. The main challenges are currently lane perception, lane tracking, and localization. However, additional research is definitely needed to address issues regarding lane change planning, particularly related to urban domains. For instance, in constrained spaces it may be necessary to find a middle ground between lateral object avoidance *within* a lane and fully changing lanes. Finally, if the focus is shifted towards an Indian metropolis, lane change behavior planning may be an entirely different issue.

Human Aspects

The focus for this thesis was automated lane changes with no interaction with a human driver except for possibly informing him or her. The test person study in the evaluation section showed that there is still a desire to have at least the *option* of commanding such lane changes. In the cockpit of the *Audi A7 piloted driving concept* vehicles there were a set of buttons to command lane changes for development purposes. For the Ur:ban project, this approach has been modified such that lane changes can be commanded. In the Stadtpilot project the indicator lever has been used to confirm lane changes (Ulbrich & Maurer, 2013). However, there is a lot of research left, such as how to design a human machine interaction for lane change behavior planning. This is even more relevant considering that strategies to bring such technologies into the hands of customers will probably start with a strong driver interaction and only later grant higher autonomy to an automated system.

16.2 Concluding Remarks

Peak of Popularity

To the author, automated driving is a challenging yet extremely interesting field of research. At the time of writing, rarely a week passes without an article about automated driving in the media. After a topic raises to the hype of its popularity and reaches its peak of inflated expectations, it often falls into a trough of disillusionment (cf. Gartner, 2015). For lane change planning, this trough of disillusionment will probably be about the numerous details that are still to be implemented and the complexity of cooperative interaction with other drivers. Thinking about traffic in an Indian metropolis or aging, poorly maintained automated vehicles on the roads pinpoints several still unaddressed challenges.

Transition

Given the public announcements of car and information technology companies, at least less sophisticated tactical behavior planning will be available to customers in a couple of years. For the author, it will be very interesting to be part of that gradual transition from human driven vehicles to possibly one day fully automated driving for everyone.

Part IV

Appendix

A Requirement Specification

A.1 Functional Requirements

Table A.1: Functional requirements

Nr.	Requirement	Passed
FR_001	The system must decide if a lane change in a given traffic situation is possible with an error rate of less than α errors per hour of the driving time.	✓/✗
FR_001_1	The system has to assess the traffic situation based on what is relevant for the decision making with an error rate of less than α errors per hour of the driving time.	✓/✗
FR_001_1_1	The system has to evaluate whether a lane change is possible based on objects on the neighbor lane behind itself. The system has to be able to handle fast objects approaching the ego vehicle from behind with at least α m/s relative velocity.	✓/✗
FR_001_1_2	The system has to evaluate whether a lane change is possible based on objects on the neighbor lane in front of itself. The system has to be able to handle fast approaches to objects with a relative velocity of at least α m/s.	✓
FR_001_1_3	The system has to evaluate whether a lane change is possible based on objects on the ego lane in front of itself. The system has to be able to detect objects in the ego lane obstructing the correct execution of a lane change with an object distance of at least α meters.	✓
FR_001_1_4	The system has to evaluate whether a lane change is possible based on fast approaching objects on the ego lane behind itself, which might be intending to overtake the ego object by changing to the neighbor lane soon.	✓

Continued on next page

Table A.1 – continued from previous page

Nr.	Requirement	Passed
FR_001_1_5	The system has to be able to consider objects directly next to it with a relative velocity of less than α m/s for lane change decision making.	✓/✗
FR_001_1_6	The system has to evaluate whether a lane change is possible based on the infrastructure situation.	✓
FR_001_1_6_1	The system has to evaluate whether a lane change is possible based on the lane marking type of the appropriate lane boundary.	✓
FR_001_1_6_2	The system has to evaluate whether a lane change is possible based on the ego and neighbor lane type. A lane change towards an emergency lane may only be allowed for a safe stop maneuver lane change.	✓
FR_001_2	The system has to be able to overturn a previously made decision about whether a lane change is possible based on more recent information within α ms.	✓
FR_002	The system must decide if a lane change is beneficial, either based on an operator's input or based on a situation assessment.	✓
FR_002_1	The system must be able to determine if an operator wishes to input whether, and if yes when, a lane change is beneficial.	✓
FR_002_2	If the system is in the automated driving mode, it has to determine if a lane change is beneficial based on the current traffic situation with an error rate of less than α errors per hour of driving time.	✓
FR_002_2_1	The system has to be able to determine whether a lane change is beneficial based on a cost/distance metric to follow the route towards the navigation destination.	✓
FR_002_2_2	The system has to be able to assess the dynamic traffic situation, whether changing the lanes would result in a dynamic benefit.	✓

Continued on next page

Table A.1 – continued from previous page

Nr.	Requirement	Passed
FR_002_2_3	The system has to be able to assess, whether a lane change is beneficial based on timing restrictions regarding prior driving events.	✓
FR_003	The system has to be able to handle uncertain and/or temporarily incorrect measurement data and to fulfill the aforementioned functional requirements. Based on these it has to derive the – towards a goal criteria – best possible decisions.	✓
FR_003_1	The system has to be able to consider the measurement uncertainties of object positions, velocities, accelerations and their existence on a particular lane.	✓
FR_003_2	The system has to be able to provide – towards its decision behavior – robustness regarding (temporarily) incorrect measurement data. A metric to quantify such temporary incorrectness has to cover the duration something is incorrect as well as the deviation from the correct value.	✓
FR_004	The system has to monitor all its input data quality and availability.	✓/✗
FR_004_1	The system has to monitor observability of the 360° field of view around the vehicle using sensor systems at all times.	✓/✗
FR_004_2	The system has to monitor and detect faults of sensors or other components at all times.	✓
FR_004_3	The system has to be able to detect delayed, distorted, or faulty sensor data at all times.	✓/✗
FR_004_3_1	The system has to be able to detect delayed sensor data at all times.	✓
FR_004_3_2	The system has to be able to detect distorted sensor data at all times.	✗

Continued on next page

Table A.1 – continued from previous page

Nr.	Requirement	Passed
FR_004_3_3	The system has to be able to detect faulty sensor data at all times.	✗
FR_005	The system has to monitor its functional skills and abilities in a given traffic situation with an error rate of less than α errors per hour of driving time.	✓
FR_005_1	The system has to be aware of its ability restrictions at all times and to be able to consider them.	✓
FR_005_1_1	The system has to be able to detect its ability restrictions to assess whether a lane change is possible when there is no speed limit and there are no objects in the neighbor lane.	✓
FR_005_1_2	The system has to be able to prevent lane changes when there is a risk of merging towards a lane directly where a highway on-ramp merges towards the highway.	✓
FR_005_1_3	The system has to be able to prevent lane changes if the curvature of the ego lane is higher than α rad/m.	✓
FR_005_1_4	The system has to be able to prevent lane changes if the ego velocity is less than $\alpha\ m/s$.	✓
FR_005_1_5	The system has to be able to prevent lane changes if the current domain type cannot be handled.	✓
FR_005_1_6	The system has to be able to prevent lane changes if an oversteering with a turning moment of more than α Nm of the driver is detected.	✓
FR_005_1_7	The system has to be able to prevent lane changes if the lateral offset to the ego lane's center is more than α m.	✓
FR_005_2	The system has to be aware of its skill restrictions at all times and has to be able to consider them.	✓
FR_005_2_1	The system has to monitor its object perception viewing range in its ego and immediate neighbor lanes.	✓

Continued on next page

Table A.1 – continued from previous page

Nr.	Requirement	Passed
FR_005_2_2	The system has to monitor its lane perception viewing range in its ego and immediate neighbor lanes.	✓
FR_006	The system has to be able to adjust the ego vehicle longitudinally towards a best gap in order to prepare a lane change in a search range of 100 m to the front and 100 *m* to the rear.	✓
FR_007	The system has to take care of flashing the indicator before initiating a lane change at least 1 second before it starts to impose a lateral displacement.	✓
FR_008	The system has to be able to detect the current domain it is in and be able to handle the following domains and transitions between them: Highway, rural road, urban areas.	✓/✗
FR_008_1	The system has to be able to perform lane changes on highways and all possible domain transitions towards and away from highways.	✓/✗
FR_008_1_1	The system has to be able to perform lane changes on a multi lane highway.	✓
FR_008_1_2	The system has to be able to perform lane changes to enter a highway on an on-ramp.	✓/✗
FR_008_1_3	The system has to be able to perform lane changes in weaving areas of highway interchanges with a length of less than 200 m.	✓/✗
FR_008_1_4	The system has to be able to perform lane changes onto off-ramps less than 100 m after the off-ramp started.	✓
FR_008_1_5	The system has to be able to change lanes in a traffic jam with a traffic flow velocity of less than 5 m/s and gaps of less than 10 m between vehicles.	✗
FR_008_2	The system has to be able to perform lane changes on rural roads and all possible domain transitions towards and away from rural roads.	✓/✗
FR_008_2_1	The system has to be able to perform lane changes on rural roads with multiple lanes in one traffic direction.	✓

Continued on next page

Table A.1 – continued from previous page

Nr.	Requirement	Passed
FR_008_2_2	The system has to be able to perform lane changes onto an oncoming lane to overtake a vehicle.	✗
FR_008_3	The system has to be able to perform lane changes in urban areas and in all possible domain transitions towards and away from urban areas.	✗
FR_009	The system has to be able to execute a safe stop lane change maneuver by continuing to perform lane changes to the right until an emergency lane is reached.	✓
FR_010	The system has to provide consistent driving orders. It has to overturn less than α decisions per hour of driving time.	✓
FR_010_1	The system has to abort flashing the indicator less than α times per hour of driving time.	✓
FR_010_2	The system has to abort an already started lane change less than α times per hour of driving time.	✓

A.2 User Interface Requirements

Table A.2: User interface requirements

Nr.	Requirement	Passed
UIR_001	The system operator must be able to override the system at all times.	✓
UIR_001_1	The system operator has to be able to prevent the vehicle from executing a lane change by holding the steering wheel with less than α Nm turning moment.	✓
UIR_001_2	The system operator has to be able to execute a lane change by turning the steering wheel with at most α Nm turning moment.	✓
UIR_002	The system has to inform the user/operator about its status and intentions at all times.	✓

A.3 Usability Requirements

Table A.3: Usability requirements

Nr.	Requirement	Passed
UR_001	Stakeholder: Safety operator → User Interface Requirements	✓
UR_002	Stakeholder: Passenger A passenger has to be able to see on a visually appealing GUI if a lane change is about to be executed and hear an indicator flashing sound.	✓
UR_003	Stakeholder: Developer	✓
UR_003_1	A developer has to be able to understand and contribute to the lane change module in less than one year.	✓
UR_003_1_1	There has to be a Doxygen documentation for each class in the productive code.	✓
UR_003_1_2	There has to be a Doxygen documentation for each function in the productive code.	✓
UR_003_1_3	Every variable in a productive function has to be named according to the coding guidelines.	✓
UR_003_1_4	50% of the productive functions' lines of code has to be comment lines.	✓
UR_003_2	A developer has to be able to log the system state, the system input data and the system outputs at all times.	✓

A.4 Performance Requirements

Table A.4: Performance requirements

Nr.	Requirement	Passed
PR_001	Data processing latency: The system may not base its current decision on sensor data older than 100 ms if newer data is available.	✓
PR_002	Computational Load: The system may not use more than 30% of the computation power of a quad-core desktop computer at any time.	✓
PR_003	Memory Load: The system may not use more than 4 GByte of RAM of a desktop computer at any time.	✓

A.5 Not Fully Met Requirements

Not fully met Requirements

Fast Objects

Which requirements are currently not yet met by the here presented implementation? For functional requirements, the requirement FR_001_1_1 and hence its superordinated requirements are not yet met. This requirement addresses fast objects on the neighbor lane approaching the ego vehicle from behind. It is not fully met because, very fast vehicles from behind may currently not be detected early enough. Workarounds to address these sensor limitations while sacrificing comfort have been introduced in section 10.3.2.

Close Objects

A second functional requirement that is not yet sufficiently met is FR_001_1_5. It addresses that objects with a minimal relative velocity right next to the ego vehicle may currently get lost by the radar sensors, which are used to monitor the ego vehicle surrounding. As pointed out in Chapter 14 and section 16.1 this is rather a shortcoming of the sensor set in the host vehicle. Industry leading near field sensor solutions in series vehicles are able to address this issue with better quality already at the time of writing. Yet, even with a best in class sensor set, this is a challenging requirement given that objects on the neighbor lane may be very small like a motor bike or even a carbon sports bicycle with little radar reflections.

Observability

Another requirement that is not yet met is FR_004_1 addressing the 360° environment observability monitoring. At the time of writing, the observability monitoring is very limited as described in section 10.3.2. No sophisticated observability grid approach is used but a rather simple ray-tracing is employed. While this works well for streets with low curvatures like highways, the limitations will easily become obvious in more demanding urban domains.

The requirements FR_004_3_2 and FR_004_3_3 are not yet met. They enforce the detection of faulty or distorted sensor data at all times. So far, no data plausibility checks have been implemented except for entirely missing sensor data. Yet, wrong sensor data like a flipped sign of an object velocity due to an e/e error or distorted sensor data due to a camera prism, which has melted in the desert sun are currently not detected by the system. For a research vehicle, it may be justified not to spend time on those seldomly occuring errors; for a series production automated vehicles these issues need to be detected by sensor data validity checks, redundancy, and cross plausibility checks among different sensor systems.

Faulty and Distorted Sensor Data

FR_008_1_2 and FR_008_1_3 address lane changes from an on-ramp onto a highway as well as lane changes in weaving areas of highway interchanges. Under favorable conditions, lane changes work in those scenarios. However, if conditions are not favorable those lane changes may fail and there may not be sufficient maneuver space left from the weaving area or the on-ramp to resolve the situation by automated maneuvers. Currently the vehicle may need a human driver to help out of these situations with his or her currently superior cooperation skills and better risk estimation skills.

On-Ramps and Weaving Areas

Likewise, on rural roads and in urban areas the cooperative skills as well as the system and environment monitoring skills of the automated vehicle are currently insufficient to handle all lane change situations (FR_008_2 and FR_008_3). Although it is possible to perform lane changes on multi-lane rural and urban roads, there are plenty of situations where the automated vehicle is currently unable to interpret and handle situations correctly in urban and rural domains. An obvious requirement that is not fulfilled are lane changed on lanes with oncoming traffic in rural domains. A human driver is able to perform those, the automated vehicle is currently too limited by its object detection and object to lane association skills (FR_008_2_2).

Rural and Urban Domains

B Description of the Audi A7 Piloted Driving Concept Vehicle

This section describes the setup of "Jack", the automated vehicle being used for the implementation and evaluation of the algorithms described in this thesis.

In January 2015, Volkswagen Group Research, Audi, and the Electronics Research Lab (ERL) of Volkswagen of America presented "Jack", the *Audi A7 piloted driving concept* vehicle for automated driving, at the Consumer Electronics Show 2015 in the US.[1] In February 2015[2] and April 2015[3], it was presented to the media on a German highway around Braunschweig and Ingolstadt. Moreover, the implementation was used for lane changes in a demonstration vehicle for the UR:BAN project[4] and as a system under test in the DADAS project for testing dependable driver assistance systems.

Figure B.1: "Jack", the *Audi A7 piloted driving concept* vehicle. Source: Audi

Base Vehicle

Two Audi A7 4G (cf. Figure B.1) and three Audi A6 4G, one Audi S6 4G, and one Volkswagen Golf are used as base vehicles. They have been modified by opening up the interfaces to the power train, the braking system, the steering, and to access information from several data buses in the vehicle. Moreover, the vehicles have been equipped with a stronger power generator and two additional batteries.

Lidar and Radar Sensors

All vehicles have been equipped with additional Radar, Lidar, and camera sensor systems as in Figure B.2. The sensor viewing ranges are depicted in Figure B.3. One front facing and one rear facing Valeo Scala B2 laser scanner are installed in the center position of the front and rear bumper and serve as a primary sensor to cover the front and rear area of the automated vehicle. They span a horizontal field of view of 145° with 4 layers and a vertical opening angle of 3.2°. According to the

[1] http://www.audi.com/content/com/brand/en/vorsprung_durch_technik/content/2014/10/piloted-driving.html visited on 05/02/2016.

[2] http://www.stern.de/auto/news/jack-das-selbstfahrende-auto-von-audi-erstmals-auf-einer-deutschen-autobahn-2174446.html visited on 05/02/2016.

[3] https://www.audi-mediaservices.com/publish/ms/content/en/public/pressemitteilungen/2015/04/10/federal_minister_of.html visited on 05/02/2016.

[4] http://urban-online.org/en/urban.html visited on 05/02/2016.

specifications, the detection range is up to 150 m for cars, approximately 200 m for trucks, and about 50 m for pedestrians under good conditions. In real world driving scenarios on a highway, typical viewing ranges are around 100 m for cars. The laser scanners are supplemented with one Bosch LRR3 long-range radar system with an opening angle of 30° horizontally and a viewing range of up to 250 m to the front and two Bosch MRR radar systems with radar beams to the front and rear respectively. The Bosch MRRs have a viewing range of about 120 m and an opening angle of 45°.

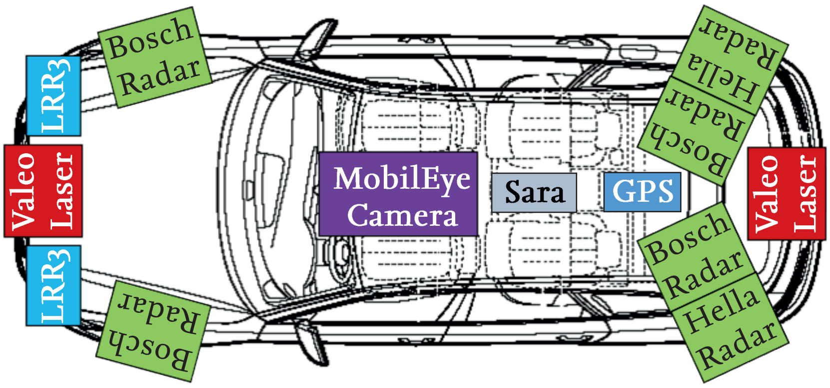


Figure B.2: Sensor setup: Sensor positions

Side Areas

Depending on the vehicle, the side areas are covered with different sensor systems. Initially, all vehicles used four Bosch MRR mid-range radar systems and two Hella SWA blind-spot radar systems as part of the regular series production blind-spot assistance. However, to improve perception skills the sensors to the side have been updated by four Delphi RSDS radar systems. Thus, at the time of writing, two vehicles already have the Delphi RSDS setup, while the remaining four still have the MRR/SWA combination. However, the evaluations were done with the series production vehicle MRR/SWA sensor setup.

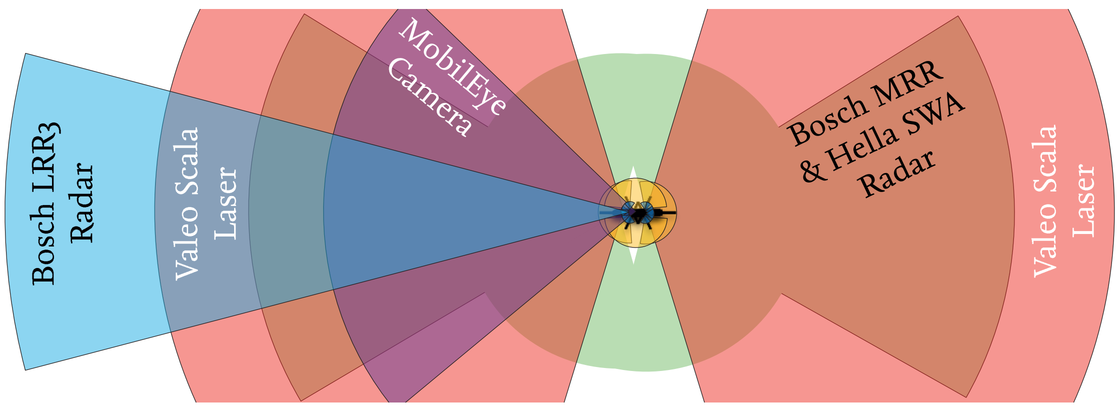


Figure B.3: Sensor setup: Sensor viewing ranges

Camera System

The setup is completed by a Kostal/MobilEye mono-vision camera system mounted instead of the series production camera behind the windscreen. It is used for lane marking detection and – to some extent – for object tracking and classification. For global satellite positioning, no expensive DGPS-RTK solution has been used. Instead, the standard Global Positioning System (GPS) antenna together with a

NovAtel OEMStar low cost, single frequency Global Navigation Satellite System (GNSS) is combined with a series production inertial measurement unit "SARA" from an Audi A8.

GPS

For parking applications, additional sensor systems have been integrated, for instance, the 360° round view camera systems or ultrasonic sensors. However, as they have not been used for lane change behavior planning, they are not discussed in detail.

Additional sensors

C State Variables in the Measurement Model

Table C.1: State variables in measurement model

Abbreviation	Variable Description
LcPos	Lane change possible
PosDyn	Lane change possible due to dynamic situation
DynNeighborRear	Lane change possible due to dynamic situation on the neighbor lane at the rear
DynNeighborFront	Lane change possible due to dynamic situation on the neighbor lane in the front
LcPosDynEgoFront	Lane change possible due to dynamic situation on the ego lane in front
PosInf	Lane change possible due to infrastructure situation
NeighLaneExists	Lane change possible due to infrastructure situation of existing neighbor lane
OnValidEgoLane	Lane change possible due to infrastructure situation of driving on a valid ego lane
AblRes	Lane change possible due to ability induced skill restrictions
RiskTooFastVehicle	Lane change possible due to ability induced skill restrictions of too fast vehicles
RiskColWiMergVeh	Lane change possible due to ability induced skill restrictions of collision with merging vehicles
CurvTooHighForLc	Lane change possible due to ability induced skill restrictions of too high curvature for a lane change
EgoTooSlowForLc	Lane change possible due to ability induced skill restrictions of the ego vehicles were too slow for lane changes

Continued on next page

Table C.1 – continued from previous page

Abbreviation	Variable Description
EnvironmentType	Lane change possible due to ability induced skill restrictions of environment type
DriverSteering	Lane change possible due to ability induced skill restrictions of detected driver steering
LatEgoOffset	Lane change possible due to ability induced skill restrictions of too high lateral offset to a lane center
SkillRes	Lane change possible due to skill restrictions
SensorViewRange	Lane change possible due to skill restrictions of sensor viewing ranges
LaneViewingRange	Lane change possible due to skill restrictions of lane viewing ranges
LcBen	Lane change beneficial
BenDyn	Lane change beneficial due to dynamic environment
EgoLaneVelocity	Lane change beneficial due to dynamic environment: Ego lane velocity
NeighborVelGain	Lane change beneficial due to dynamic environment: Neighbor lane velocity gain
AccFollow	Lane change beneficial due to dynamic environment: Ego vehicle is in ACC follow mode
RLDrivingOrder	Lane change beneficial due to dynamic environment: Ego vehicle is in right lane driving order
PressingRearVehicle	Lane change beneficial due to dynamic environment: Ego vehicle is followed by pressing rear vehicle
TimRes	Lane change beneficial due to timing restrictions
TiRestrAfterAnyLc	Lane change beneficial due to a timing restriction after any lane change
TiRestrAfterLcL	Lane change beneficial due to a timing restriction after a lane change left
TiRestrAfterPresVeh	Lane change beneficial due to a timing restriction after the detection of a pressing vehicle
BenInf	Lane change beneficial due to infrastructure

Continued on next page

Table C.1 – continued from previous page

Abbreviation	Variable Description
InfraBeneGain	Lane change beneficial due to infrastructure benefit gain
EgoLaneAdvice	Lane change beneficial due to ego lane advice
PredictiveIndicator	Lane change beneficial due to predictive indicator activation
LGapQualRear2	Left gap quality of 2nd rear gap
LGapQualRear1	Left gap quality of 1st rear gap
LGapQual0	Left gap quality of neighbor gap
LGapQualFront1	Left gap quality of 1st front gap
LGapQualFront2	Left gap quality of 2nd front gap
RGapQualRear2	Right gap quality of 2nd rear gap
RGapQualRear1	Right gap quality of 1st rear gap
RGapQual0	Right gap quality of neighbor gap
RGapQualFront1	Right gap quality of 1st front gap
RGapQualFront2	Right gap quality of 2nd front gap
Left	Lane change status left
Right	Lane change status right
NormalDriving	Lane change status of normal driving
LcInProg	Lane change in progress
LcInPrep	Lane change in preparation
LcIndic	Lane change indicated
LcAbort	Lane change in abortion

D Coordinate System

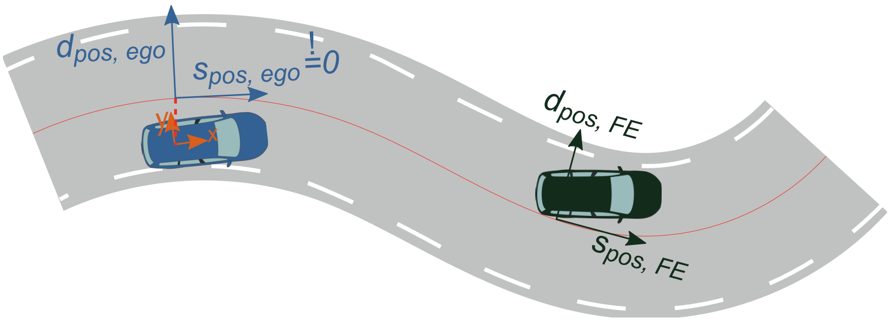


Figure D.1: Coordinate system. x/y is the ego vehicle coordinate system tied to the center of the ego vehicle (blue) at the rear axle directly above the ground. s_{pos} is the longitudinal distance of an element along the lane reference path (red, solid). The point of origin is moved with the projected point of origin of the ego coordinate system onto the reference path. d_{pos} is the lateral distance of an element to the reference path. The reference path changes to a neighbor lane if the ego vehicle changes its lane. s_{vel}/d_{vel} is the absolute longitudinal/lateral velocity of an element in relation to the (stationary) ground. s_{acc}/d_{acc} is the absolute longitudinal/lateral acceleration. Their orientation is along the s_{pos}/d_{pos} coordinates. This resembles the coordinate system in Werling (2010, p. 30) with the ego coordinate system's origin according to the department's definition at Volkswagen.

E Necessary Rear Sensor Viewing Range

This attachment calculates necessary sensor viewing ranges to detect and track objects to the rear. A scenario is assumed where an automated vehicle got stuck behind a truck driving with 80 km/h on the second fastest lane of a highway. The scenario may occur on a German highway without a speed limit where overtaking is only allowed on the left.

Table E.1: Parameters viewing range calculation

Parameter	Value	Description
$s_{vel,ego}$	$22.22\,\mathrm{m/s}$	Assumed ego velocity
$s_{vel,RN}$	$\Delta s_{vel} + s_{vel,ego}$	Velocity of a fast approaching rear vehicle on the left neighbor lane
$s_{acc,RN}$	-1, -2, -4, -7, or $-10\,\mathrm{m/s^2}$	Braking deceleration of fast approaching rear neighbor vehicle
T_R	$1\,\mathrm{s}$	Driver reaction time of fast approaching rear neighbor vehicle
T_{RN}	0.8 or 1.8 s	Minimal time gap to be left in front of the fast approaching rear neighbor vehicle

Table E.1 provides parameters to be used in equation 10.6 in Chapter 10. Figure E.1 illustrates the necessary viewing ranges for dynamic elements to the rear as a function of their velocity. If lane changes shall be performed on a multi-lane highway and shall not be blocked whenever there is a fast vehicle on *any* lane, it will be necessary to detect and track objects with a lateral accuracy of the lane width at the distances calculated in Figure E.1. Considering technological limitations and issues like achievable calibration accuracies, this is very challenging for Lidar sensors as well as Radar sensors.

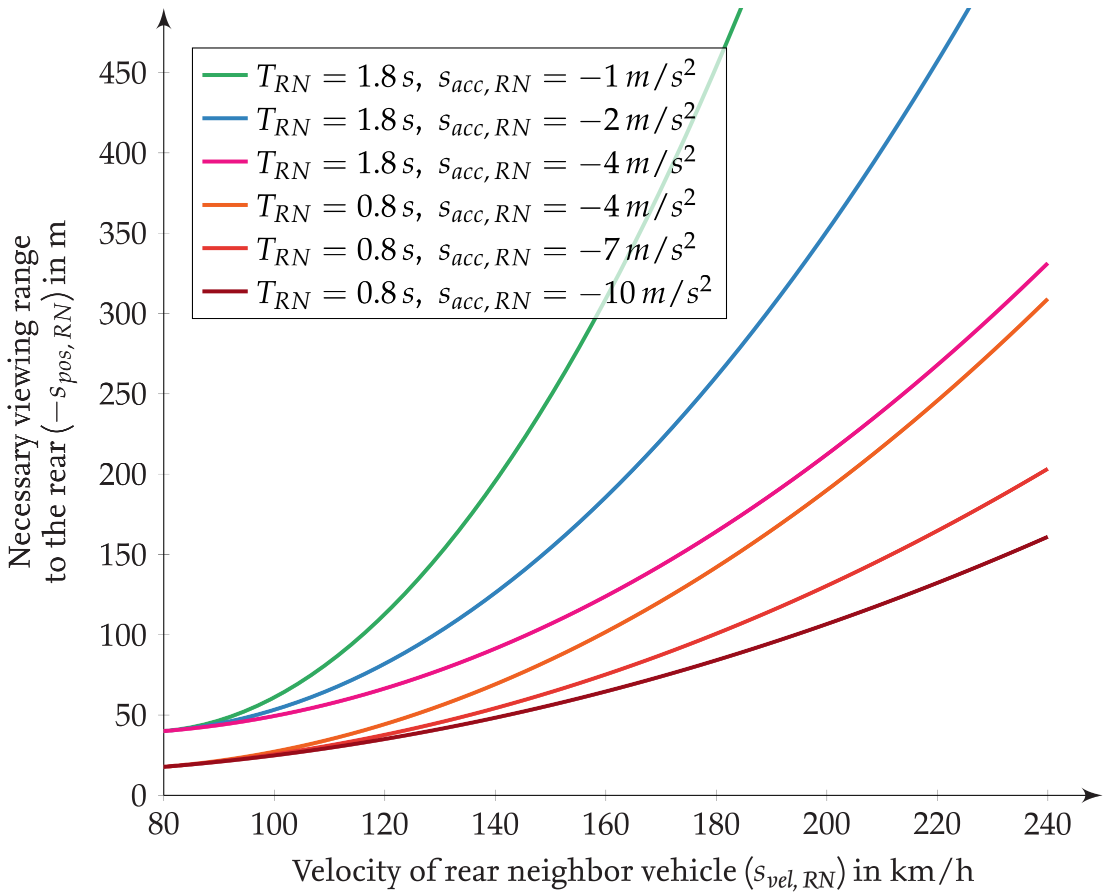


Figure E.1: Necessary dynamic element detection and tracking range as a function of the rear neighbor vehicle's velocity approaching an ego vehicle with $s_{vel,ego} = 80\,\text{km/h}$

F Lane Change Benefit Transfer Functions

F.1 Dynamic Benefit Transfer Function

```
Input: Traffic velocity on ego and neighbor lane
Result: Dynamic lane change benefit
Function CalcDynamicBenefit(EgoLaneVelocity, NeighborLaneVelocity) is
    // Parameters:
    yMinBeneficialVal = 0.35
    yMaxBeneficialVal = 1.2
    xBeginOfNegativeVelGain = 0.0
    xNoMoreNegativeVelGain = −3.5
    xBeginOfPositiveVelGain = 3.5
    xNoMorePositiveVelGain = 8.0
    neutralRes = 0.5
    Res = yMinBeneficialVal
    velocityGain = NeighborLaneVelocity - EgoLaneVelocity
    if velocityGain >= xNoMoreNegativeVelGain then
        Res = yMinBeneficialVal +
                (neutralRes - yMinBeneficialVal)*
                (velocityGain - xNoMoreNegativeVelGain)/
                    (xBeginOfNegativeVelGain - xNoMoreNegativeVelGain)
    end
    if velocityGain >= xBeginOfNegativeVelGain then
        Res = neutralRes
    end
    if velocityGain >= xBeginOfPositiveVelGain then
        Res = neutralRes +
                (yMaxBeneficialVal - neutralRes)*
                (velocityGain - xBeginOfPositiveVelGain)/
                    (xNoMorePositiveVelGain - xBeginOfPositiveVelGain)
    end
    if velocityGain >= xNoMorePositiveVelGain then
        Res = yMaxBeneficialVal
    end
    return Res
end
```

Algorithm 1: Algorithm for dynamic lane change benefit calculation

F.2 Infrastructure Lane Advice Transfer Function

```
Input: Lane advice of ego and neighbor lane
Result: Infrastructure-related lane change benefit
Function CalcInfrastructureBenefit(EgoLaneAdvice, NeighborLaneAdvice) is
    // Parameters:
    LaneAdviceHysteresis = 0.01
    NeutralValue = 0.5
    NeutralThreshold = 0.075
    if NeighborLaneAdvice+LaneAdviceHysteresis >= EgoLaneAdvice then
        NeighborLaneAdvice = 1.0
        Denominator = 1.0+EgoLaneAdvice
    else
        Denominator = 1.0+NeighborLaneAdvice
    end
    Benefit = NeighborLaneAdvice/Denominator
    if abs(Benefit-NeutralValue) < NeutralThreshold then
        Benefit = NeutralValue
    end
    return Benefit
end
```

Algorithm 2: Algorithm for infrastructure-related lane change benefit calculation

G Lane Change Reward Functions

```
Function cRewardModel::evaluate(state=b(t_n), action=u(t_n)) is
    // Parameters:
    reward=vector⟨tFloat64⟩(eReDim_NumOfDimensions,0)
    // Calculate in which planning level we currently are in the decision tree.
    level = (int) (action.duration/planningTimestep -1);
    // Calculate the sign for the hysteresis
    hystSignL = 0.0; hystSignR=0.0
    determineHysteresisSign(state, hystSignL, hystSignR);
    // Use gap qualities as they are
    reward[LGapQual0:LGapQual4] = state.hiddenStateVars.mu[LGapQualRear2:LGapQualFront2]
    reward[RGapQual0:RGapQual4] = state.hiddenStateVars.mu[RGapQualRear2:RGapQualFront2]
    // Take the value minus the threshold +/- hysteresis => range:[-0.5 .. 0.5]
    state.hiddenStateVars.mu[LLcPos] = transfLcPos(state.hiddenStateVars.mu[LLcPos], hystSignL)
    state.hiddenStateVars.mu[RLcPos] = transfLcPos(state.hiddenStateVars.mu[RLcPos], hystSignR)
    state.hiddenStateVars.mu[LLcBen] = transfLcBen(state.hiddenStateVars.mu[LLcBen], hystSignL)
    state.hiddenStateVars.mu[RLcBen] = transfLcBen(state.hiddenStateVars.mu[RLcBen], hystSignR)
    fLLcBenWhenLcWasInitiated =
        transfLcBen(state.hiddenStateVars.mu[LLcBenWhenLcWasInitiated], hystSignL)
    fRLcBenWhenLcWasInitiated =
        transfLcBen(state.hiddenStateVars.mu[RLcBenWhenLcWasInitiated], hystSignR)

    switch state.status do
        case eLcStatus_LcNormalDriving
                || eLcStatus_LcInPreparationL || eLcStatus_LcInPreparationR
            CalcRewardNormalDriving()
        case eLcStatus_LcIndicatedL
            CalcRewardIndicatingLaneChangeLeft()
        case eLcStatus_LcInProgL
            CalcRewardLaneChangeInProgressLeft()
        case eLcStatus_LcIndicatedR
            // [...] code for indicating a lane change right is
            //         identical to lane change left except being mirrored [...]
            // Hint: tBool bPlanLcRight = (state.hiddenStateVars.mu[LLcBen]
            //                              < state.hiddenStateVars.mu[RLcBen])
        case eLcStatus_LcInProgR
            // [...] code for executing a lane change right is
            //         identical to lane change left except being mirrored [...]

    // Normalize toward duration
    reward[eReDim_Possible] = reward[eReDim_Possible] * action.duration/planningTimestep
    reward[eReDim_Beneficial] = reward[eReDim_Beneficial] * action.duration/planningTimestep
    // If a lane change is to be initiated, compensate the gain just caused by
       getting to the other side of the hysteresis
    CompensateGainInRewardByJustCrossingHysteresis()
    return reward
```

Algorithm 3: Algorithm for reward calculation

```
Function cRewardModel::calcRewardNormalDriving(state=b(t_n), action=u(t_n)) is
    if (action.action == eLcActionIndicateLcL) then
        reward[eReDim_Possible] = rewardMultiplier * state.hiddenStateVars.mu[LLcPos]
        if (state.hiddenStateVars.mu[LLcPosAbilityRestr] < 0.1)
            || (state.hiddenStateVars.mu[LLcPosSkillRestr] < 0.1) then
            reward[eReDim_Possible] += m_params.reward.rewardDoSkillOrAbilityRestrictedAction
        reward[eReDim_Beneficial] = rewardMultiplier * state.hiddenStateVars.mu[LLcBen]
        // if predictive indicator mode is requested, give a high reward for
           suggesting a lane change
        if (state.hiddenStateVars.mu[LLcBenInfraPredictiveIndicator]>0.5) && (level == 0) then
            reward[eReDim_Possible] += m_params.reward.rewardFlashPredictiveIndicator
            reward[eReDim_Beneficial] += m_params.reward.rewardFlashPredictiveIndicator
        if (state.status == eLcStatus_LcInPreparationL) then
            // if LcPrepSlowdown. Override the fact that it is currently not possible
               to change lanes.
            tBool bPlanLcLeft = (state.hiddenStateVars.mu[LLcBen]
                                > state.hiddenStateVars.mu[RLcBen])
            if ((state.hiddenStateVars.mu[LcPrepSlowdown] > m_params.reward.lcPrepSlowdownThres)
                    && bPlanLcLeft && (level == 0)) then
                reward[eReDim_Possible] += m_params.reward.rewardFlashGapAdjIndicator
    else if (action.action == eLcActionIndicateLcR) then
        // [...] code for indicating a lane change right is identical to lane
           change left except being mirrored [...]
    else if (action.action == eLcActionDriveNormal || action.action == eLcActionAbortPrepLc
            || action.action == eLcActionAbortSuggestLc) then
        // check if a lane change to the left or to the right is better and take
           the better one of both the decision reference.
        reward[eReDim_Possible] =
            -max(rewardMultiplier *
                        min(state.hiddenStateVars.mu[LLcPos], state.hiddenStateVars.mu[LLcBen]),
                rewardMultiplier *
                        min(state.hiddenStateVars.mu[RLcPos], state.hiddenStateVars.mu[RLcBen]))
        reward[eReDim_Beneficial] = reward[eReDim_Possible]
        // compare with the minimum of left and right. This takes the better
           alternative (whether left or right) as a benchmark
        // then we take the negative of it, because drive normal means, we will not
           execute that action.
    continued on next page
```

Algorithm 4: Algorithm for calcRewardNormalDriving()

```
else if (action.action == eLcActionPrepareLcL) then
    // No matter what, take the neutral solution as a starting point
    reward[eReDim_Possible] =
        -max(rewardMultiplier *
                    min(state.hiddenStateVars.mu[LLcPos], state.hiddenStateVars.mu[LLcBen]),
            rewardMultiplier *
                    min(state.hiddenStateVars.mu[RLcPos], state.hiddenStateVars.mu[RLcBen]))
    reward[eReDim_Beneficial] = reward[eReDim_Possible]
    // Do longitudinal planning
    tFloat64 comfortReward = 0
    if (action.deltaVel< 0) then
        // drive slower
        comfortReward = action.deltaVel * m_params.reward.rewardDriveSlower
    else
        // drive faster
        comfortReward = action.deltaVel * m_params.reward.rewardDriveFaster
    // ONLY for preparing a lane change right eLcActionPrepareLcR, not left
    // comfortReward = std::max(−10.0,comfortReward);
    tFloat64 gapAdjustmentReward = 10
    tFloat64 bestGapQuality = state.hiddenStateVars.mu[LGapQualityRear0]
    bestGapQuality = max(bestGapQuality,state.hiddenStateVars.mu[LGapQualityFront2])
    bestGapQuality = max(bestGapQuality,state.hiddenStateVars.mu[LGapQualityFront1])
    bestGapQuality = max(bestGapQuality,state.hiddenStateVars.mu[LGapQualityRear1])
    bestGapQuality = max(bestGapQuality,state.hiddenStateVars.mu[LGapQualityRear2])
    static const tFloat64 gapQualityMultiplier =
        m_params.reward.gapAdjRewardMultiplier*rewardMultiplier
    gapAdjustmentReward =
        max(gapAdjustmentReward, bestGapQuality*gapQualityMultiplier)
    if (state.hiddenStateVars.mu[LLcPos]< 0 && state.hiddenStateVars.mu[LLcBen]> 0) then
        reward[eReDim_Possible] += comfortReward + gapAdjustmentReward
        reward[eReDim_Beneficial] += comfortReward + gapAdjustmentReward
    else
        // penalize to continue doing PrepLc, when a lane change is possible
           already
        reward[eReDim_Possible] += m_params.reward.rewardRemainInStatePrepareLc
        reward[eReDim_Beneficial] += m_params.reward.rewardRemainInStatePrepareLc
else if (action.action == eLcActionPrepareLcR) then
    // [...] code for preparing a lane change right is identical to lane change
       left except being mirrored
    // and comfortReward = std::max(−10.0,comfortReward) to limit the comfort
       penalties at highway exits
else
    reward[eReDim_Possible] = m_params.reward.rewardImpossibleAction
    reward[eReDim_Beneficial] = m_params.reward.rewardImpossibleAction
return reward
```

Algorithm 5: Algorithm for calcRewardNormalDriving()

```
Function cRewardModel::calcRewardIndicatingLaneChangeLeft(state=b(t_n), action=u(t_n)) is
    // if 3sec=>1.0; if 2sec=>1.0; if 1sec=>0.5; if 0sec=>0.0
    tFloat64 tooFastMulti = min(max(((state.timeSinceLastStatusChange-2.0)/2.0+1.0),0.0),1.0)
    // Do not take min(state.hiddenStateVars.mu[LLcPos],
       state.hiddenStateVars.mu[LLcBen]) but only
       state.hiddenStateVars.mu[LLcPos] because we do not want to abort a lane
       change due to a low Benefit
    if (action.action == eLcActionDoLcL) then
        reward[eReDim_Possible] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcPos] * tooFastMulti
        reward[eReDim_Beneficial] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcBen] * tooFastMulti
    else if (action.action == eLcActionIndicateLcL) then
        reward[eReDim_Possible] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcPos] * (1-tooFastMulti)
        reward[eReDim_Beneficial] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcBen] * (1-tooFastMulti)
        // If predictive indicator mode is requested, give a high reward for (keep)
           indicating a lane change
        if ((state.hiddenStateVars.mu[LLcBenInfraPredictiveIndicator]> 0.5)
              && (state.hiddenStateVars.mu[LLcPos]< 0.0) then
            reward[eReDim_Possible] += m_params.reward.rewardFlashPredictiveIndicator
            reward[eReDim_Beneficial] += m_params.reward.rewardFlashPredictiveIndicator
        // if LcPrepSlowdown. Override the fact that it is currently not possible
           to change lanes and continue indicating
        tBool bPlanLcLeft = (state.hiddenStateVars.mu[LLcBen] > state.hiddenStateVars.mu[RLcBen])
        if (((state.hiddenStateVars.mu[LcPrepSlowdown] > m_params.reward.lcPrepSlowdownThres)
        || (state.timeSinceLastLcPrepSlowdown < maxIndLaneChangeDurationAfterLcPrepSlowdown))
              && bPlanLcLeft && (state.hiddenStateVars.mu[LLcPos]< 0.0) && (level == 0)) then
            reward[eReDim_Possible] += m_params.reward.rewardFlashGapAdjIndicator
            // override beneficial because (1-tooFastMulti) converges to zero after a
               long time
            reward[eReDim_Beneficial] += m_params.reward.rewardFlashGapAdjIndicator
    continued on next page
```

Algorithm 6: Algorithm for calcRewardIndicatingLaneChangeLeft()

```
else if (action.action == eLcActionAbortLcL) then
    reward[eReDim_Possible] = -rewardMultiplier * state.hiddenStateVars.mu[LLcPos]
    reward[eReDim_Beneficial] = -rewardMultiplier *
     max(fLLcBenWhenLcWasInitiated,state.hiddenStateVars.mu[LLcBen])
    // if predictive indicator mode is requested, give a high reward for
       suggesting a lane change
    if ((state.hiddenStateVars.mu[LLcBenInfraPredictiveIndicator]> 0.5) && (level==0)) then
        reward[eReDim_Possible] -= m_params.reward.rewardFlashPredictiveIndicator
        reward[eReDim_Beneficial] -= m_params.reward.rewardFlashPredictiveIndicator
    // if LcPrepSlowdown. Override the fact that it is currently not possible
       to change lanes and continue indicating
    tBool bPlanLcLeft = (state.hiddenStateVars.mu[LLcBen] > state.hiddenStateVars.mu[RLcBen])
    if (((state.hiddenStateVars.mu[LcPrepSlowdown] > m_params.reward.lcPrepSlowdownThres)
      || (state.timeSinceLastLcPrepSlowdown < maxIndLaneChangeDurationAfterLcPrepSlowdown))
        && bPlanLcLeft && (state.hiddenStateVars.mu[LLcPos]< 0.0) && (level == 0)) then
        reward[eReDim_Possible] -= m_params.reward.rewardFlashGapAdjIndicator
        // Override beneficial because (1-tooFastMulti) converges to zero after a
           long time
        reward[eReDim_Beneficial] -= m_params.reward.rewardFlashGapAdjIndicator
else
    reward[eReDim_Possible] = m_params.reward.rewardImpossibleAction
    reward[eReDim_Beneficial] = m_params.reward.rewardImpossibleAction
return reward
```

Algorithm 7: Algorithm for calcRewardIndicatingLaneChangeLeft()

```
Function cRewardModel::calcRewardLaneChangeInProgressLeft(state=b(t_n), action=u(t_n)) is
    // if 4sec=>1.0; if 3sec=>1.0; if 2sec=>0.66; if 1sec=>0.33; if 0sec=>0.0
    tFloat64 tooFastMulti = min(max(((state.timeSinceLastStatusChange-3.0)/3.0+1.0),0.0),1.0)
    // Do not take min(state.hiddenStateVars.mu[LLcPos],
       state.hiddenStateVars.mu[LLcBen]) but only
       state.hiddenStateVars.mu[LLcPos] because we do not want to abort a lane
       change due to a low Benefit
    if (action.action == eLcActionFinishLcL) then
        reward[eReDim_Possible] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcPos] * tooFastMulti
        reward[eReDim_Beneficial] =
            rewardMultiplier *
         max(fLLcBenWhenLcWasInitiated,state.hiddenStateVars.mu[LLcBen]) * tooFastMulti
        if (state.hiddenStateVars.mu[LLcPos] < −0.4) then
            reward[eReDim_Possible] += m_params.reward.rewardCollisionPenalty
            reward[eReDim_Beneficial] += m_params.reward.rewardCollisionPenalty
    else if (action.action == eLcActionDoLcL) then
        reward[eReDim_Possible] =
            rewardMultiplier * state.hiddenStateVars.mu[LLcPos] * (1-tooFastMulti)
        reward[eReDim_Beneficial] = rewardMultiplier
            * max(fLLcBenWhenLcWasInitiated,state.hiddenStateVars.mu[LLcBen]) *
         (1-tooFastMulti)
        if (state.hiddenStateVars.mu[LLcPos] < −0.4) then
            reward[eReDim_Possible] += m_params.reward.rewardCollisionPenalty
            reward[eReDim_Beneficial] += m_params.reward.rewardCollisionPenalty
    else if (action.action == eLcActionAbortLcL) then
        reward[eReDim_Possible] = -rewardMultiplier * state.hiddenStateVars.mu[LLcPos]
        reward[eReDim_Beneficial] =
            -rewardMultiplier *
         max(fLLcBenWhenLcWasInitiated,state.hiddenStateVars.mu[LLcBen])
    else
        reward[eReDim_Possible] = m_params.reward.rewardImpossibleAction
        reward[eReDim_Beneficial] = m_params.reward.rewardImpossibleAction
    return reward
```

Algorithm 8: Algorithm for calcRewardLaneChangeInProgressLeft()

```
Function cRewardModel::compensateGainInRewardByJustCrossingHysteresis(state=b(t_n), action=u(t_n)) is
    // If a lane change is to be initiated, compensate the gain just caused by
       getting to the other side of the hysteresis
    if ((state.status == eLcStatus_LcNormalDriving || state.status == eLcStatus_LcInPreparationL ||
     state.status == eLcStatus_LcInPreparationR) && (action.action == eLcActionIndicateLcL || action.action
     == eLcActionIndicateLcR)) then
        // For all future time steps we will make it by
           hysteresisLcPos/hysteresisLcBen easier to make a lane change in order to
           avoid instabilities. Hence, even if the benefit is totally constant and
           slightly below the decision threshold, we will still initiate a lane
           change because future situations will consider the same benefit value
           more beneficial because of the hysteresis. To compensate this, we need
           to increase the costs of initiating a lane change up front. Why
           depth=level+1?: We need to start compensating for the next time level
           Why m_params.reward.planningDepth-2?: The last two actions are for any
           really reasonable policy to finish a lane change and to drive normal on
           the new lane.
        for (int depth=level+1;depth<m_params.reward.planningDepth−2;depth++) do
            // Calculate the discounting factor
            tFloat64 gamma = pow(discount,depth)
            // Calculate the duration of that situation
            tFloat64 fduration = planningTimestep*(depth+1)
            // Compensate the possible hysteresis. Consider the gamma, the duration
               and the rewardMultiplier as it is done anyways
            reward[eReDim_Possible] += -gamma * fduration/planningTimestep
                                 * rewardMultiplier * m_params.reward.hysteresisLcPos
            // Compensate the beneficial hysteresis. Consider the gamma, the duration
               and the rewardMultiplier as it is done anyways. Multiply by
               normalizeBenefitFactor because it is done for the rewardBeneficial,
               too.
            reward[eReDim_Beneficial] += -gamma * fduration/planningTimestep
                 * rewardMultiplier * m_params.reward.hysteresisLcBen*m_normalizeBenefitFactor
    return reward
```

Algorithm 9: Algorithm for compensateGainInRewardByJustCrossingHysteresis()

```
Function cRewardModel::transfLcPos(val, hysteresisSign) is
    return (val − (m_params.reward.thresholdLcPos
          + hysteresisSign*m_params.reward.hysteresisLcPos))
```

Algorithm 10: Algorithm for transformLcPossible()

```
Function cRewardModel::transfLcBen(val, hysteresisSign) is
    return min((val - (m_params.reward.thresholdLcBen
        + hysteresisSign*m_params.reward.hysteresisLcBen))
        *m_normalizeBenefitFactor,0.5)
```

Algorithm 11: Algorithm for transformLcBeneficial()

```
Function cRewardModel::determineHysteresisSign(state=b(t_n), out hystSignL, out hystSignR) is
    // calculates the hysteresisSign for lane change pos/ben hysteresis
       left/right
    if ((state.status == eLcStatus_LcNormalDriving || state.status == eLcStatus_LcInPreparationL ||
     state.status == eLcStatus_LcInPreparationR)) then
        // if a lane change has not yet been started, take a higher decision
           threshold
        hystSignL = +1.0
        hystSignR = +1.0
    else if ((state.status == eLcStatus_LcIndicatedL) || (state.status == eLcStatus_LcSuggestedL) ||
     (state.status == eLcStatus_LcInProgL)) then
        // if a lane change has been started, take a lower decision threshold
        hystSignL = -1.0
        // if a lane change has not yet been started, take a higher decision
           threshold
        hystSignR = +1.0
    else if ((state.status == eLcStatus_LcIndicatedR) || (state.status == eLcStatus_LcSuggestedR) ||
     (state.status == eLcStatus_LcInProgR)) then
        // if a lane change has not yet been started, take a higher decision
           threshold
        hystSignL = +1.0
        // if a lane change has been started, take a lower decision threshold
        hystSignR = -1.0
```

Algorithm 12: Algorithm for determineHysteresisSign()

Table G.1: Parameters for reward evaluation

Parameter	Value	Description
rewardDriveFaster	−3	Penalty for deviating from adaptive cruise control target velocity
rewardDriveSlower	−3	Penalty for deviating from adaptive cruise control target velocity
rewardRemainInStatePrepareLc	−30	Penalty for only preparing a lane change
rewardFlashPredictiveIndicator	1000	Reward for flashing the indicator if we are in a situation for preparing a lane change
rewardFlashGapAdj—Indicator	100	Reward for flashing the indicator to squeeze into a gap
rewardImpossibleAction	−10000	Penalty for planning a physically impossible action
rewardCollisionPenalty	−1000	Penalty for intentionally planning a collision
rewardDoSkillOrAbility—RestrictedAction	−700	Penalty for doing something where ego vehicle's skills/abilities are insufficient for
gapAdjRewardMultiplier	0.15	Multiplier to weight gap adjustments
thresholdLcPos	0.5	Threshold for when a lane change is considered possible
thresholdLcBen	0.27	Threshold for when a lane change is considered beneficial
normalizeBenefitFactor	2.0	Normalize range of benefit values
lcPrepSlowdownThres	0.5	Threshold if the ego vehicle should do a slowdown to prepare a lane change
planningDepth	7	Number of steps to plan into the future
rewardMultiplier	100	Multiplier to weight rewards
discount	0.5	Discount factor to discount future rewards
planningTimestep	0.5 s	Default planning timestep size
maxIndLaneChangeDuration—AfterLcPrepSlowdown	5.6 s	How long does the ego vehicle accept to indicate a lane change without being able to execute it after it has actively reduced the velocity to align to a gap

H Questionnaire for Test Person Study [1]

Probandennummer:
Datum:

Auto-Pilot Realfahrt-Studie

Vorbefragung

Demographische Daten

1. Geschlecht — männlich ☐ weiblich ☐
2. Alter — ______ Jahre

Fahrverhalten

3. Wie oft fahren Sie Auto?
   - Nie ☐
   - Selten ☐
   - Mehr als einmal im Monat ☐
   - Mehr als einmal in der Woche ☐
   - Fast täglich ☐

4. Fahren Sie lieber selbst oder lassen Sie jemand anderes fahren?
   - Ich fahre lieber selbst ☐
   - Ich lasse jemand anderes fahren ☐

   Warum? ____________________

5. Wie würden Sie Ihren eigenen Fahrstil einschätzen?

   ☐ sehr defensiv ☐ eher defensiv ☐ eher sportlich ☐ sehr sportlich

6. Wie gut spiegeln die folgenden Aussagen Ihre Meinung wider? Bitte geben Sie Ihre Zustimmung auf der Skala von ***'stimmt gar nicht'*** bis ***'stimmt völlig'*** an.

	Stimmt gar nicht	**Stimmt eher nicht**	**teils / teils**	**Stimmt eher**	**Stimmt völlig**
Ich bin ein besserer Autofahrer im Vergleich zum allgemeinen Durchschnitt.	☐	☐	☐	☐	☐
Mein Unfall- und Gefahren-Risiko während des Autofahrens ist geringer als der Durchschnitt.	☐	☐	☐	☐	☐
Ich kann besser mit Gefahrensituationen im Straßenverkehr umgehen als der Durchschnitt.	☐	☐	☐	☐	☐

Figure H.1: Questionnaire page 1

[1]This questionnaire has been designed together with my colleagues Amelie Stephan and Ina Othersen.

Probandennummer:
Datum:

Erfahrung mit Fahrassistenzsystemen

7. Haben Sie Erfahrung mit den folgenden Fahrerassistenzsystemen?

	Keine Erfahrung	Wenig Erfahrung	Gewöhnt an System	Ständige Nutzung	Kenne ich nicht
CC – Cruise Control Sorgt automatisch für eine frei wählbare konstante Geschwindigkeit	☐	☐	☐	☐	☐
ACC – Adaptive Cruise Control Geschwindigkeit und Distanz zum voraus-fahrenden Fahrzeug wird automatisch gehalten	☐	☐	☐	☐	☐
HC – Heading Control Automatische Spurhaltung; der Fahrer wird durch regulierende Eingriffe unterstützt	☐	☐	☐	☐	☐

Figure H.2: Questionnaire page 2

Probandennummer:
Datum:

Umgang mit Technik

8. Wie gut spiegeln die folgenden Aussagen ihre Meinung wider? Bitte geben Sie Ihre Zustimmung auf der Skala von **'trifft gar nicht zu'** bis **'trifft voll zu'** an.

	trifft gar nicht zu	trifft nicht zu	teils / teils	trifft eher zu	trifft voll zu
Es fällt mir leicht, die Bedienung eines elektronischen Geräts zu lernen.	☐	☐	☐	☐	☐
Ich kenne mich im Bereich elektronischer Geräte aus.	☐	☐	☐	☐	☐
Elektronische Geräte führen zu geistiger Verarmung.	☐	☐	☐	☐	☐
Ich informiere mich über elektronische Geräte, auch wenn ich keine Kaufabsicht habe.	☐	☐	☐	☐	☐
Elektronische Geräte machen vieles umständlicher.	☐	☐	☐	☐	☐
Es macht mir Spaß, ein elektronisches Gerät auszuprobieren.	☐	☐	☐	☐	☐
Ich habe bzw. hätte Verständnisprobleme beim Lesen von Elektronik und Computerzeitschriften.	☐	☐	☐	☐	☐
Elektronische Geräte erleichtern mir den Alltag.	☐	☐	☐	☐	☐
Ich bin begeistert, wenn ein neues elektronisches Gerät auf den Markt kommt.	☐	☐	☐	☐	☐
Elektronische Geräte verringern den persönlichen Kontakt zwischen den Menschen.	☐	☐	☐	☐	☐
Ich kenne die meisten Funktionen der elektronischen Geräte, die ich besitze.	☐	☐	☐	☐	☐
Elektronische Geräte erhöhen die Sicherheit.	☐	☐	☐	☐	☐
Elektronische Geräte machen krank.	☐	☐	☐	☐	☐
Elektronische Geräte helfen, an Informationen zu gelangen.	☐	☐	☐	☐	☐
Ich gehe gern in den Fachhandel für elektronische Geräte.	☐	☐	☐	☐	☐
Elektronische Geräte verursachen Stress.	☐	☐	☐	☐	☐
Elektronische Geräte ermöglichen einen hohen Lebensstandard.	☐	☐	☐	☐	☐
Ich liebe es, neue elektronische Geräte zu besitzen.	☐	☐	☐	☐	☐
Elektronische Geräte machen unabhängig.	☐	☐	☐	☐	☐

Figure H.3: Questionnaire page 3

Probandennummer:
Datum:

Bewertung Fahrt 4

19. Bitte beantworten Sie die folgende Frage auf der Skala von **1 – 'sehr gering'** bis **15 – 'sehr hoch'**.

sehr gering			gering			neutral			hoch			sehr hoch		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Wie würden Sie die **Fahrleistung** des hochautomatisch fahrenden Fahrzeugs **bei Fahrstreifenwechseln** bewerten? ☐

Wie würden Sie das **Fahrstreifenwechsel-Verhalten** des hochautomatisch fahrenden Fahrzeugs einschätzen?
☐ sehr defensiv ☐ eher defensiv ☐ eher sportlich ☐ sehr sportlich

20. Wie gut spiegeln die folgenden Aussagen Ihre Meinung wider? Bitte geben Sie Ihre Zustimmung auf der Skala von ***'stimmt gar nicht'*** bis ***'stimmt völlig'*** an.

	Stimmt gar nicht	**Stimmt eher nicht**	**teils / teils**	**Stimmt eher**	**Stimmt völlig**
Ich hätte mir gewünscht, dass das hochautomatisch fahrende Fahrzeug bereits bei kleineren individuellen Nachteilen einen Fahrstreifenwechsel beginnt	☐	☐	☐	☐	☐
Ich bewerte die Motivation des hochautomatisch fahrenden Fahrzeugs zur Durchführung von Fahrstreifenwechseln als angemessen.	☐	☐	☐	☐	☐
Ich bewerte die Auswahl geeigneter Lücken für Fahrstreifenwechsel als angemessen für ein hochautomatisch fahrendes Fahrzeug.	☐	☐	☐	☐	☐
Ich bewerte die Durchführung von Fahrstreifenwechseln als angemessen für ein hochautomatisch fahrendes Fahrzeug.	☐	☐	☐	☐	☐
Das Fahrstreifenwechsel-Verhalten ist in seiner Defensivität / Sportlichkeit angemessen für ein hochautomatisch fahrendes Fahrzeug.	☐	☐	☐	☐	☐
Ich hätte gerne die Möglichkeit gehabt, dem hochautomatisch fahrenden Fahrzeug selbst taktische Fahrentscheidungen vorzugeben.	☐	☐	☐	☐	☐
Abbrüche von bereits begonnenen Fahrstreifenwechseln habe ich als störend erlebt.	☐	☐	☐	☐	☐

Figure H.4: Questionnaire page 4

Probandennummer:
Datum:

Abschlussbefragung

Gesamtbewertung

21. Bitte beantworten Sie die folgende Frage auf der Skala von **1 – 'sehr gering'** bis **15 – 'sehr hoch'**.

sehr gering			gering			neutral			hoch			sehr hoch		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Wie würden Sie die erlebte **Fahrleistung** des Auto-Piloten insgesamt bewerten?	
Wie **nervös** haben Sie sich insgesamt während der Fahrsituationen gefühlt?	
Wie gut konnten Sie das Verhalten des hochautomatisch fahrenden Fahrzeugs insgesamt **vorhersagen**?	
Wie sehr konnten Sie sich insgesamt darauf **verlassen**, dass das hochautomatisch fahrende System funktioniert?	
Wie hoch ist insgesamt Ihr **Glaube** daran, dass das hochautomatisch fahrende Fahrzeug mit jeder zukünftigen Situation im eben erlebten Fahrkontext umgehen kann?	
Wie hoch ist Ihr **Vertrauen** in das hochautomatisch fahrende System insgesamt?	

22. Würden Sie das hochautomatische System gern in Ihrem eigenen Fahrzeug haben?
ja ☐ nein ☐

23. Gab es Situationen während der hochautomatischen Fahrt, in denen Sie sich unwohl gefühlt haben oder lieber selbst die Kontrolle über das Fahrzeug übernommen hätten?

___

24. Gab es Informationen, die Sie in dieser Situation vermisst haben und im Anzeigekonzept **benötigt** hätten?

___

___

___

25. Gab es Informationen im Anzeigekonzept, die Sie in dieser Situation als **überflüssig** empfunden haben?

___

___

___

Figure H.5: Questionnaire page 5

I Publications Affiliated with the Author

I.1 Peer-Reviewed Articles

Choi, J., Ulbrich, S., Lichte, B. & Maurer, M. (Oct. 2013). “Multi-Target Tracking using a 3D-Lidar sensor for autonomous vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 881–886.

Matthaei, R., Reschka, A., Bagschik, G., Escher, M., Menzel, T., Rieken, J., Scheide, T., Schuldt, F., Ulbrich, S., Wendler, J. T., Hecker, P. & Maurer, M. (2015a). “Das Projekt Stadtpilot - Automatisiertes Fahren an der TU Braunschweig (English Translation: The Stadtpilot Project - Automated Driving at the TU Braunschweig)”. In: *ZfAW - Zeitschrift für die gesamte Wertschöpfungskette*, vol. 18, no. 1, pp. 12–23.

Matthaei, R., Reschka, A., Rieken, J., Dierkes, F., Ulbrich, S., Winkle, T. & Maurer, M. (2015b). “Autonomes Fahren (English Translation: Autonomous Driving)”. In: *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Ed. by Winner, H., Hakuli, S., Lotz, F. & Singer, C. 3rd. Wiesbaden, Germany: Springer Vieweg, pp. 1139–1165.

Matthaei, R., Reschka, A., Rieken, J., Dierkes, F., Ulbrich, S., Winkle, T. & Maurer, M. (2016). “Autonomous Driving”. In: *Handbook of Driver Assistance Systems - Basic Information, Components and Systems for Active Safety and Comfort*. Ed. by Winner, H., Hakuli, S., Lotz, F. & Singer, C. Springer, pp. 1519–1556. English version of Matthaei et al. (2015).

Reschka, A., Bagschik, G., Ulbrich, S., Nolte, M. & Maurer, M. (2015). “Ability and skill graphs for system modeling, online monitoring, and decision support for vehicle guidance systems”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 933–939.

Ulbrich, S. & Maurer, M. (2013). “Probabilistic online POMDP decision making for lane changes in fully automated driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 2063–2067.

Ulbrich, S., Nothdurft, T., Maurer, M. & Hecker, P. (June 2014). “Graph-based context representation, environment modeling and information aggregation for automated driving”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 541–547.

Ulbrich, S., Großjohann, S., Appelt, C., Homeier, K., Rieken, J. & Maurer, M. (2015a). “Structuring Cooperative Behavior Planning Implementations for Automated Driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 2159–2165.

Ulbrich, S. & Maurer, M. (2014). “Evaluation einer taktischen Verhaltensentscheidungsfindung für Fahrstreifenwechsel beim vollautomatisierten Fahren in Städten (English Translation: Evaluation of a Tactical Behavior Decision Making Algorithm for Lane Changes in Fully Automated Driving in Cities)”. In: *Workshop Fahrerassistenzsysteme (FAS 2014)*, pp. 147–158.

Ulbrich, S. & Maurer, M. (2015a). “Situation Assessment in Tactical Lane Change Behavior Planning for Automated Vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 975–981.

Ulbrich, S. & Maurer, M. (2015b). “Towards Tactical Lane Change Behavior Planning for Automated Vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 989–995.

Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F. & Maurer, M. (2015b). “Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 982–988.

Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F. & Maurer, M. (2015c). “Definition der Begriffe Szene, Situation und Szenario für das automatisierte Fahren (English Translation: Defining the Terms Scene, Situation and Scenario for Automated Driving)”. In: *Workshop Fahrerassistenzsysteme (FAS 2015)*, 105–117, German Version of Ulbrich et al. (2015b).

Ulbrich, S., Reschka, A., Rieken, J., Ernst, S., Bagschik, G., Dierkes, F., Nolte, M. & Maurer, M. (2017a). *Towards a Functional System Architecture for Automated Vehicles*. URL: `https://arxiv.org/abs/1703.08557`.

Ulbrich, S., Schuldt, F., Homeier, K., Steinhoff, M., Menzel, T., Krause, J. & Maurer, M. (2017b). “Testing and Validating Tactical Lane Change Behavior Planning for Automated Driving”. In: *Automated Driving - Safer and more efficient future driving*. Ed. by Horn, M. & Watzenig, D. Springer International Publishing AG, pp. 451–471.

I.2 Patent Applications and Disclosures of Inventions

Appelt, C., Großjohann, S., Homeier, K. & Ulbrich, S. (2015). “Projektion des Verkehrsumfelds auf drei parallele Fahrstreifen (English Translation: Projection of the Traffic Environment onto Three Parallel Lanes)”. Disclosure of Invention (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015a). “Ermitteln von Fahrstreifenflussgeschwindigkeiten zur Bewertung des Nutzens von Fahrstreifenwechseln in dünnem Verkehr (English Translation: Determining Lane Specific Traffic Flow Velocities to Evaluate the Benefit of Performing Lane Changes in Sparse Traffic)”. Disclosure of Invention (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015b). “Kommunikation einer Fahrstreifenwechselabsicht (English Translation: Communication of a Lane Change Intention)”. Patent DE 102015219934 A1 (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015c). “Marginale, integrale Vorteilhaftigkeit bei der Bewertung des Überholnutzens eines geringfügig langsameren Vorderfahrzeugs (English Translation: Marginal, Integral Benefit for Evaluating the Overtaking Utility of a Marginally Slower Front Vehicle)”. Pending Patent Application (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015d). “Verfahren und Vorrichtung in einem Kraftfahrzeug zum automatisierten Fahren (English Translation: Method and Device in a Motor Vehicle for an Automatic Drive)”. Patent DE 102015217498 A1 (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015e). “Verfahren zur Befolgung des Rechtsfahrgebots bei der Bewertung des Nutzens von Fahrstreifenwechseln (English Translation: Procedure for Adherence of the Right Lane Driving Order Rule in the Benefit Evaluation of Lane Changes)”. Pending Patent Application (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015f). “Verfahren zur Berücksichtigung der Bedeutung von hintereinander fahrenden Objekten (English Translation: Procedure to Consider the Significance of in Succession Driving Vehicles)”. Disclosure of Invention (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015g). “Verfahren zur Längsausrichtung an Lücken für Fahrstreifenwechsel und Lückenbewertung (English Translation: Procedure for Longitudinal Gap Adjustment for Lane Changes and Gap Quality Assessment)”. Disclosure of Invention (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2016). “Verfahren und Vorrichtung in einem Kraftfahrzeug zum automatisierten Fahren (English Translation: Method and Device in a Motor Vehicle for an Automatic Drive)”. International Patent Application for DE 102015217498 A1: WO 2017/046101 A1 (Germany).

I.3 Supervised Theses

Arens, D. (2014). “Funktionserweiterung eines Visualisierungsmoduls für Objekthypothesen in der Umfeldrepräsentation automatischer Straßenfahrzeuge”. Bachelor Thesis. TU Braunschweig.

Kakar, J. (2013). “Implementing and Evaluating a Split-and-Merge Segmentation Algorithm for Object Detection in LIDAR Point Cloud Data”. Bachelor Thesis. TU Braunschweig.

Löbke, H. (2014). "Implementierung eines Konzepts zur online Trajektoriengenerierung & -stabilisierung in zeitkritischen Verkehrsszenarien". Master Thesis. TU Braunschweig.

Matthaei, J. (2012). "Entwurf einer probabilistischen Umfeldwahrnehmung und Objektverfolgung". Master Thesis. TU Braunschweig.

I.4 Contributions in the Media (TV and Radio)

Date	Medium	Contribution
11.03.2012	ARD & KiKA	Die Sendung mit der Maus: Auto mit Autopilot
24.11.2012	KiKA	Motzgurke.tv: Die Tigerentenreporter zeigen's euch!
20.02.2013	NDR Info	LOGO - Wissenschaft aus Braunschweig: Das vernetzte Auto
11.08.2013	KiKA	Erde an Zukunft
29.08.2013	SRF	Einstein
12.09.2013	3sat	Nano
28.11.2013	BR	BR-Zündfunk
12.12.2013	NDR	Hallo Niedersachsen
21.12.2013	ORF1	Genauso geht's
09.11.2014	NDR	Hallo Niedersachsen: Ehemalige Grenzregion: Vom Zonenrandgebiet zur Boom-Region
06.05.2015	HR	Alles Wissen: Zukunft der Mobilität: Automatisiertes Fahren
02.06.2015	RTL Nord	Demografiekongress in Hannover
18.10.2015	RTL II	Echtzeit: Autonomes Auto
10.12.2015	WDR	Planet Wissen: Auto der Zukunft

Contributions in print and online media are excluded due to excessive length and incomplete tracking. Contributions about the Audi A7 piloted driving concept vehicle are excluded because of incomplete tracking.

J Glossary

Consistency

Is understood such that a decision should fit in the framework of the decisions being taken so far. In Psychology, behavioral consistency is defined by Häcker (2014, p. 868) how human behavior remains constant (or relatively constant) among situations (transsituative consistency) or over time (temporal stability). He defines behavioral coherency if a behavior is licit and predictable without behavioral consistency. In game theory (e.g., Nau & McCardle, 1990), coherent behavior means "not presenting opportunities for arbitrage [...] to an outside observer who serves as betting opponent." In linguistics, a set of propositions are consistent if they are free of contradiction. A set of propositions is coherent, if they are linked with each other.

Context

The context is a general phrase to describe the *part of discourse that surrounds and represents an element.* This entails the scene with all its components. Above this, the context is assumed to span at least the live time of the element. Thus, it entails many scenarios. Due to this wide definition, a full context may never be represented in its entirety.

Domain

According to Pellkofer (2003, p. 5), a domain is the class of environments, in which whole groups of actions are allowed, necessary, or forbidden. To the author, the domain is part of the context. It summarizes aspects that distinguish, e.g., an urban domain from a highway domain. Yet, to the author a domain is rather a set of aspects that hold true for a certain amount of time but do not necessarily need to span the lifetime of an element.

Dynamic element

Deviating from Geyer et al.'s (2014) definition of "*dynamic elements*" being based on the temporal extent of their scene definition, *dynamic elements* are assumed to move (having kinetic energy), or possibly being able to move (having sufficient energy and abilities to move). Past movements (object has stopped at traffic light) are a strong indicator for potential movements in the immediate future. Current perception skills of technical systems are not sufficient to classify stationary elements as *dynamic*, therefore a statue anchored to the ground may currently not be differentiable from a not moving pedestrian. Hence, a pedestrian may possibly be misclassified as being part of the scenery, or a statue as being part of the dynamic elements. According to section 2.3, the author does not distinguish objects from subjects, because the automated vehicle is not able to distinguish them either. If necessary dynamic elements are distinguished between actors and observers.

Environment

Similar to scene except for the considered element's self representation. Thus the environment is everything that surrounds the considered element.

Harm

Other than in the ISO 26262, the term *harm* is not only limited to "physical injury or damage to the health of persons" (ISO, 2011, Part I, p. 9), but also entails "material damage" and "actual or potential ill effects or danger" as in the Oxford dictionaries Oxford (2015b).

Maneuver

The term maneuver is used to describe a sequence of (tactical) actions. Typical maneuvers in the field of automated driving are for instance overtaking another vehicle or performing a lane change. For a detailed discussion, the reader is referred to section 2.4.

Map

Based on Oxford (2017a) and Ulbrich et al. (2015g), a map is a diagrammatic representation of an area's scenery.

Motivation

Motivation is used to address "the motivational processes involved in goal setting" (Achtziger & Gollwitzer, 2008, p. 276).

Risk

According to the ISO 26262, *risk* is the "combination of the probability of occurrence of harm and the severity of that harm" (ISO, 2011, Part I, p. 13). Due to a wider definition of harm, the risk definition got extended, as well.

Scenario

A scenario describes the temporal development between several scenes in a sequence of scenes. Every scenario starts with an initial scene. Actions & events as well as goals & values may be specified to characterize this temporal development in a scenario. Other than a scene, a scenario spans a certain amount of time.

Scene

A scene describes a snapshot of the environment including the scenery and dynamic elements, as well as all actors' and observers' self-representations, and the relationships among those entities. Only a scene representation in a simulated world can be all-encompassing (objective scene, ground truth). In the real world it is incomplete, incorrect, uncertain, and from one or several observers' points of view (subjective scene).

Scenery

The *scenery* subsumes all geo-spatially stationary aspects of the scene. This entails metric, semantic and topologic information about roads and all their components like lanes, lane markings, road surfaces, or the roads' domain types. Moreover, this subsumes information about conflict areas between lanes as well as information about their interconnections, e.g., at intersections. Apart from the before mentioned environment conditions, the scenery also includes stationary elements like houses, fences, curbs, trees, traffic lights, or traffic signs.

Self-representation

A *self-representation* contains the current skill levels and general system skills as well as the states and attributes of an actor or observer. The skills may be represented in a very basic form like a timeout signal from a sensor system or in a sophisticated form of a skill graph as proposed by Reschka et al. (2015). For observers, the field of view and occlusions are an essential part of its skills. The actors'/observers' states and attributes entail information about the position relative to the road network, dynamic motion information, and even information from the (vehicle's) data busses like whether an indicator is currently activated or not.

Severity

According to the ISO 26262, *severity* is the "estimate of the extent of harm to one or more individuals that can occur in a potentially hazardous situation" (ISO, 2011, Part I, p. 16).

Situation

A situation is the entirety of circumstances, which are to be considered for the selection of an appropriate behavior pattern at a particular point of time[1]. It entails all relevant conditions, options and determinants for behavior[2]. A situation is derived from the scene by an information selection and augmentation process based on transient (e.g. mission-specific) as well as permanent goals and values. Hence, a situation is always subjective by representing an element's point of view.

Utility

According to Merriam-Webster (2015d), *utility* is "The quality or state of being useful. Thus, it subsumes anything that is favorable to the user or the system as whole."

[1] Cf. Wershofen & Graefe (1996).
[2] Cf. Meyer (1977). Determinants as in determining factors.

Value

The term “value” is used according to Kluckhohn (1965, p. 395) cited by Six (2015) as follows: “A value is a conception, explicit or implicit, distinctive of an individual or characteristic of a group, of the desirable which influences the selection from available modes, means, and ends of action.”

Volition

Based on Lewin (1926) and Ach (1935), volition is understood as “the form of motivation involved in goal striving”, thus the “translation of existing goals into action[s] and [...] the regulation of these processes” (Achtziger & Gollwitzer, 2008, p. 276). Vice versa, motivation is used to address “the motivational processes involved in goal setting” (Achtziger & Gollwitzer, 2008, p. 276).

K Bibliography

Ach, N. (1935). “Analyse des Willens (English Translation: Analysis of Will)”. In: *Handbuch der biologischen Arbeitsmethoden*. Ed. by Abderhalden, E. Berlin, Germany: Urban & Schwarzberg.

Achtziger, A. & Gollwitzer, P. M. (2008). “Motivation and Volition in the Course of Action”. In: *Motivation and Action*. Ed. by Heckhausen, J. & Heckhausen, H. Cambridge, USA: Cambridge University Press, pp. 272–295.

Aeberhard, M., Rauch, S., Bahram, M., Tanzmeister, G., Thomas, J., Pilat, Y., Homm, F., Huber, W. & Kaempchen, N. (2015). “Experience, Results and Lessons Learned from Automated Driving on Germany’s Highways”. In: *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 42–57.

Albus, J. S. (Aug. 1979). “Mechanisms of planning and problem solving in the brain”. In: *Mathematical Biosciences*, vol. 45, no. 3, pp. 247–293.

Althoff, M. (2010). “Reachability Analysis and its Application to the Safety Assessment of Autonomous Cars”. Dissertation. TU München.

Ammann, P. & Offutt, J. (2008). *Introduction to Software Testing*. Cambridge, United Kingdom: Cambridge University Press.

Anderson, S. L. (2011). “The Unacceptability of Asimov’s Three Laws of Robotics as a Basis for Machine Ethics”. In: *Machine Ethics*. Ed. by Anderson, M. & Anderson, S. L. Cambridge, United Kingdom: Cambridge University Press. Chap. 16, pp. 285–296.

Angenendt, W. (1987). *Situationsbezogene Sicherheitskriterien im Strassenverkehr (English Translation: Situation-Related Safety Criteria in Road Traffic)*. Bergisch Gladbach, Germany: Bundesanstalt für Straßenwesen (BASt).

Ardelt, M., Coester, C. & Kaempchen, N. (Dec. 2012). “Highly Automated Driving on Freeways in Real Traffic Using a Probabilistic Framework”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1576–1585.

Ardelt, M. & Waldmann, P. (2011). “Hybrides Steuerungs- und Regelungskonzept für das hochautomatisierte Fahren auf Autobahnen (English Translation: Hybrid Control Concept for Highly Automated Driving on Freeways)”. In: *at - Automatisierungstechnik*, vol. 59, no. 12, pp. 738–750.

Asimov, I. (1942). “Runaround”. In: *Astounding Science Fiction*, vol. 29, no. 1, pp. 94–103.

Asimov, I. (Jan. 1985). *Robots and Empire*. London, United Kingdom: Collins.

Autonomos (2016). *Automated Driving Challenge – 2400km in Mexico*. URL: `https://www.autonomos-systems.de/en/mexico/` (visited on 02/12/2017).

Bacha, A., Bauman, C., Faruque, R., Fleming, M., Terwelp, C., Reinholtz, C., Hong, D., Wicks, A., Alberi, T., Anderson, D., Cacciola, S., Currier, P., Dalton, A., Farmer, J., Hurdus, J., Kimmel, S., King, P., Taylor, A., Covern, D. V. & Webster, M. (Aug. 2008). "Odin: Team VictorTango's entry in the DARPA Urban Challenge". In: *Journal of Field Robotics*, vol. 25, no. 8, pp. 467–492.

Badino, H., Franke, U. & Mester, R. (2007). "Free Space Computation Using Stochastic Occupancy Grids and Dynamic Programming". In: *Proceedings International Conference on Computer Vision (ICCV)*, w/o page numbers.

Bahram, M., Wolf, A., Aeberhard, M. & Wollherr, D. (June 2014). "A prediction-based reactive driving strategy for highly automated driving function on freeways". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 400–406.

Baker, C. R. & Dolan, J. M. (Sept. 2008). "Traffic interaction in the urban challenge: Putting boss on its best behavior". In: *International IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, pp. 1752–1758.

Bandyopadhyay, T., Won, K. S., Frazzoli, E., Hsu, D., Lee, W. S. & Rus, D. (2013). "Intention-Aware Motion Planning". In: *Algorithmic Foundations of Robotics: Proceedings of the 10th Workshop on the Algorithmic Foundations of Robotics*. Springer International Publishing, pp. 475–491.

Barber, D. (2012). *Bayesian Reasoning and Machine Learning*. Cambridge, United Kingdom: Cambridge University Press.

Baron, J. & Spranca, M. (Apr. 1997). "Protected Values". In: *Organizational Behavior and Human Decision Processes*, vol. 70, no. 1, pp. 1–16.

Behere, S. & Törngren, M. (2015). "A functional architecture for autonomous driving". In: *Proceedings of the First International Workshop on Automotive Software Architecture*, pp. 3–10.

Bellman, R. (1957). *Dynamic Programming*. Princeton, USA: Princeton University Press.

Bender, P., Tas, O. S., Ziegler, J. & Stiller, C. (June 2015). "The combinatorial aspect of motion planning: Maneuver variants in structured environments". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1386–1392.

Bergholz, P. A. (2003). "Bewegungsfertigkeiten im Sportunterricht (English Translation: Motoric Skills in Sports Lessons". Dissertation. Universität Tübingen.

Bergmiller, P. (2015). *Towards Functional Safety in Drive-by-Wire Vehicles*. Berlin, Germany: Springer International Publishing.

Berman, B. (2015). *Drivers Push Tesla's Autopilot Beyond Its Abilities*. MIT Technology Review. URL: http://www.technologyreview.com/news/542651/drivers-push-teslas-autopilot-beyond-its-abilities/ (visited on 10/22/2015).

Blackmore, L. (2006). "A probabilistic particle control approach to optimal, robust predictive control". In: *AIAA Conference on Guidance, Navigation and Control*, w/o page numbers.

Borrelli, F., Bemporad, A. & Morari, M. (Feb. 2015). *Predictive Control for linear and hybrid systems*. Edition v.5 from October 2015. URL: http://www.mpc.berkeley.edu/mpc-course-material/.

Brackstone, M. & McDonald, M. (1999). "Car-following: A historical review". In: *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 2, no. 4, pp. 181–196.

Bratman, M. E. (1987). *Intention, Plans and Practical Reason*. Cambridge, USA: Harvard University Press.

Brechtel, S., Gindele, T. & Dillmann, R. (Oct. 2014). "Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 392–399.

Brechtel, S., Gindele, T. & Dillmann, R. (Oct. 2011). "Probabilistic MDP-behavior planning for cars". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1537–1542.

Brilon, W. & Brannolte, U. (1977). *Simulationsmodell für den Verkehrsablauf auf zweispurigen Straßen mit Gegenverkehr (English Translation: Simulation Model for Traffic on Two-Lane Streets with Oncoming Traffic)*. Bonn, Germany: Bundesministerium für Verkehr.

Broggi, A., Cerri, P., Debattisti, S., Laghi, M., Medici, P., Panciroli, M. & Prioletti, A. (June 2014). "PROUD-Public road urban driverless test: Architecture and results". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 648–654.

Broggi, A., Buzzoni, M., Debattisti, S., Grisleri, P., Laghi, M. C., Medici, P. & Versari, P. (Sept. 2013). "Extensive Tests of Autonomous Driving Technologies". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1403–1415.

Brooks, R. (1986). "A robust layered control system for a mobile robot". In: *IEEE Journal on Robotics and Automation*, vol. 2, no. 1, pp. 14–23.

Brown, P. J. (1995). "The stick-e document: a framework for creating context-aware applications". In: *Electronic Publishing*, vol. 8, no. 2 & 3, pp. 259–272.

Browne, C. B., Powley, E., Whitehouse, D., Lucas, S. M., Cowling, P. I., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S. & Colton, S. (Mar. 2012). "A Survey of Monte Carlo Tree Search Methods". In: *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43.

Bunge, M. (1979). *Treatise on Basic Philosophy*. Dordrecht, The Netherlands: Reidel Publishing.

Camacho, E. F., Bordons, C., Grimble, M. J. & Johnson, M. A. (2007). *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. London, United Kingdom: Springer International Publishing.

Carroll, J. B. (Jan. 1993). *Human Cognitive Abilities: A Survey of Factor-Analytic Studies*. Cambridge, United Kingdom: Cambridge University Press.

Chandler, R. E., Herman, R. & Montroll, E. W. (Apr. 1958). “Traffic Dynamics: Studies in Car Following”. In: *Operations Research*, vol. 6, no. 2, pp. 165–184.

Chen, J. (2009). “Fahrerassistenzsystem zum autonomen Spurwechsel (English Translation: Advanced Driver Assisstance System for Autonomous Lane Changes)”. Dissertation. RWTH Aachen.

Chryssanthacopoulos, J. P. & Kochenderfer, M. J. (Oct. 2011). “Analysis of open-loop and closed-loop planning for aircraft collision avoidance”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 212–217.

Clarke, R. (2011). “Asimov’s laws of robotics: implications for information technology”. In: *Machine Ethics*. Ed. by Anderson, M. & Anderson, S. L. Cambridge, United Kingdom: Cambridge University Press. Chap. 15, pp. 254–284.

Collins (2017). *Systemic*. Collins Dictionary. URL: `https://www.collinsdictionary.com/dictionary/english/systemic`.

Culligan, K. F. (2006). “Online Trajectory Planning for UAVs Using Mixed Integer Linear Programming”. Dissertation. University of London.

Dehghani, M., Tomai, E., Forbus, K. D. & Klenk, M. (2008). “An Integrated Reasoning Approach to Moral Decision-Making”. In: *AAAI Conference on Artificial Intelligence*, pp. 1280–1286.

Dey, A. K. (Jan. 2001). “Understanding and Using Context”. In: *Personal Ubiquitous Computing*, vol. 5, no. 1, pp. 4–7.

Dey, A. K. & Abowd, G. (2000). “Towards a Better Understanding of Context and Context-Awareness”. In: *Conference on Human Factors in Computing Systems*, pp. 304–307.

Dickmanns, E., Behringer, R., Dickmanns, D., Hildebrandt, T., Maurer, M., Thomanek, F. & Schiehlen, J. (Oct. 1994). “The seeing passenger car ’VaMoRs-P’”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 68–73.

Dickmanns, E. & Müller, N. (May 1996). “Scene recognition and navigation capabilities for lane changes and turns in vision-based vehicle guidance”. In: *Control Engineering Practice*, vol. 4, no. 5, pp. 589–599.

Dickmanns, E. D. (2007). *Dynamic Vision for Perception and Control of Motion*. London, United Kingdom: Springer International Publishing.

Dickmanns, E. D. & Graefe, V. (Dec. 1988). "Dynamic monocular machine vision". In: *Machine Vision and Applications*, vol. 1, no. 4, pp. 223–240.

Donges, E. (Aug. 1999). "A Conceptual Framework for Active Safety in Road Traffic". In: *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 32, no. 2-3, pp. 113–128.

Doran, J. E., Franklin, S., Jennings, N. R. & Norman, T. J. (1997). "On cooperation in multi-agent systems". In: *The Knowledge Engineering Review*, vol. 12, no. 03, pp. 309–314.

Doshi, A. & Trivedi, M. (2011). "Tactical driver behavior prediction and intent inference: A review". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1892–1897.

Du, J., Masters, J. & Barth, J. (2004). "Lane-level positioning for in-vehicle navigation and automated vehicle location (AVL) systems". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 35–40.

Düring, M. & Pascheka, P. (June 2014). "Cooperative decentralized decision making for conflict resolution among autonomous agents". In: *International IEEE Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, pp. 154–161.

Earl, M. & D'Andrea, R. (Dec. 2005). "Iterative MILP methods for vehicle-control problems". In: *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1158–1167.

Eco, U. (Feb. 1972). *Einführung in die Semiotik (English Translation: Introduction to Semiotics)*. Trans. by Trabant, J. 9th edition. Munich, Germany: Wilhelm Fink Verlag.

Eidehall, A. & Petersson, L. (Mar. 2008). "Statistical Threat Assessment for General Road Scenes Using Monte Carlo Sampling". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 1, pp. 137–147.

Elfes, A. (June 1987). "Sonar-based real-world mapping and navigation". In: *IEEE Journal on Robotics and Automation*, vol. 3, no. 3, pp. 249–265.

Enders, W. (2015). *Applied Econometric Time Series*. 4th edition. Hoboken, USA: John Wiley and Sons.

Erlemann, K. (2007). "Objektorientierte mikroskopische Verkehrssimulation (English Translation: Object-Oriented, Microscopic Traffic Simulation)". Dissertation. Ruhr-Universität Bochum.

Felix Lotz (2017). "Eine Referenzarchitektur für die assistierte und automatisierte Fahrzeugführung mit Fahrereinbindung (English Translation: A Reference Architecture for Assisted and Automated Vehicle Guidance with Driver Interaction)". Dissertation. TU Darmstadt.

Fischer, J., Gruden, S., Imhof, E. & Strub, J.-D. (2007). *Grundkurs Ethik: Grundbegriffe philosophischer und theologischer Ethik (English Translation: Basic Course Ethics: Basic Terminology of Philosopical and Theological Ethics)*. Stuttgart, Germany: Kohlhammer.

Frese, C. (2012). “Planung kooperativer Fahrmanöver für kognitive Automobile (English Translation: Planning of Cooperative Driving Maneuvers for Cognitive Vehicles)”. Dissertation. Karlsruher Institut für Technologie.

Freyer, J. (2008). *Vernetzung von Fahrerassistenzsystemen zur Verbesserung des Spurwechselverhaltens von ACC (English Translation: Interconnecting Advanced Driver Assistance Systems to Improve the Lane Change Behavior of ACC)*. Göttingen, Germany: Culliver Verlag.

Fritz, H. (2011). “Verfahren zur Unterstützung eines Fahrers bei einem Fahrspurwechsel und Vorrichtung zur Durchführung des Verfahrens (English Translation: Procedure to Support a Driver in a Lane Change and Apparatus for Executing that Procedure)”. Patent Disclosure DE102011016770A1 (Germany).

Gartner (2015). *Gartner’s 2015 Hype Cycle for Emerging Technologies*. Press Release. URL: `http://www.gartner.com/newsroom/id/3114217` (visited on 11/01/2015).

Gasser, T., Arzt, C., Ayoubi, M., Bartels, A., Eier, J., Flemisch, F., Häcker, D., Hesse, T., Huber, W., Lotz, C., Maurer, M., Ruth-Schumacher, S., Schwarz, J. & Vogt, W. (2012). *Rechtsfolgen zunehmender Fahrzeugautomatisierung (English Translation: Legal Consequences of Increasing Vehicle Automation)*. Bergisch Gladbach, Germany: Bundesanstalt für Straßenwesen (BASt).

Gazis, D. C., Herman, R. & Rothery, R. W. (Aug. 1961). “Nonlinear Follow-the-Leader Models of Traffic Flow”. In: *Operations Research*, vol. 9, no. 4, pp. 545–567.

Geiger, A., Lenz, P. & Urtasun, R. (June 2012). “Are we ready for autonomous driving? The KITTI vision benchmark suite”. In: *International IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354–3361.

Gerdes, J. C. & Thornton, S. M. (2016). “Implementable Ethics for Autonomous Vehicles”. In: *Autonomous Driving*. Ed. by Maurer, M., Gerdes, J. C., Lenz, B. & Winner, H. Berlin, Germany: Springer International Publishing, pp. 87–102.

Geyer, S., Baltzer, M., Franz, B., Hakuli, S., Kauer, M., Kienle, M., Meier, S., Weissgerber, T., Bengler, K., Bruder, R., Flemisch, F. & Winner, H. (May 2014). “Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance”. In: *IET Intelligent Transport Systems*, vol. 8, no. 3, pp. 183–189.

Gigerenzer, G. (2014). “Entscheiden unter Unsicherheit (English Translation: Decision Making under Uncertainty)”. In: *Dorsch Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 12/30/2014.

Gindele, T., Brechtel, S. & Dillmann, R. (Oct. 2013). “Learning context sensitive behavior models from observations for predicting traffic situations”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1764–1771.

Gindele, T., Brechtel, S. & Dillmann, R. (Sept. 2010). “A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments”. In:

International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 1625–1631.

Gipps, P. G. (Apr. 1981). “A behavioural car-following model for computer simulation”. In: *Transportation Research Part B: Methodological*, vol. 15, no. 2, pp. 105–111.

Gipps, P. G. (Oct. 1986). “A model for the structure of lane-changing decisions”. In: *Transportation Research Part B: Methodological*, vol. 20, no. 5, pp. 403–414.

Gips, J. (2011). “Towards the Ethical Robot”. In: *Machine Ethics*. Ed. by Anderson, M. & Anderson, S. L. Cambridge, United Kingdom: Cambridge University Press, pp. 244–253.

Go, K. & Carroll, J. M. (Nov. 2004). “The Blind Men and the Elephant: Views of Scenario-based System Design”. In: *Interactions*, vol. 11, no. 6, pp. 44–53.

Goodall, N. J. (2014). “Machine Ethics and Automated Vehicles”. In: *Road Vehicle Automation*. Ed. by Meyer, G. & Beiker, S. Springer International Publishing, pp. 93–102.

Grancharova, A. & Johansen, T. A. (2012). *Explicit Nonlinear Model Predictive Control*. Vol. 429. Lecture Notes in Control and Information Sciences. Berlin, Germany: Springer International Publishing.

Grau, C. (2011). “There is No ’I’ in ’Robot’: Robots and Utilitarianism (Expanded & Revised)”. In: *Machine Ethics*. Ed. by Anderson, S. & Anderson, M. Cambridge, United Kingdom: Cambridge University Press. Chap. 25, pp. 451–463.

Gregor, R., Lutzeler, M., Pellkofer, M., Siedersberger, K.-H. & Dickmanns, E. (Mar. 2002). “EMS-Vision: a perceptual system for autonomous vehicles”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 48–59.

Gregor, R. (2002). “Fähigkeiten zur Missionsdurchführung und Landmarkennavigation (English Translation: Abilities for Mission Execution and Landmark Navigation)”. Dissertation. Universität der Bundeswehr in München.

Grüne, L. & Pannek, J. (2011). *Nonlinear Model Predictive Control*. Communications and Control Engineering. London, United Kingdom: Springer International Publishing.

Haag, M. (1998). “Bildfolgenauswertung zur Erkennung der Absichten von Strassenverkehrsteilnehmern (English Translation: Image Sequence Evaluation for Intention Detection of Road Traffic Participants)”. Dissertation. Universität Karlsruhe.

Habenicht, S. (2012). “Entwicklung und Evaluation eines manöverbasierten Fahrstreifenwechselassistenten (English Translation: Development and Evaluation of a Maneuver-Based Lane Change Assistance System)”. Dissertation. TU Darmstadt.

Häcker, H. (2014). “Konsistenz des Verhaltens (English Translation: Consistency of Behavior)”. In: *Dorsch Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/17/2014.

Häcker, H. (2015). “Fähigkeit (English Translation: Ability)”. In: *Dorsch Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 07/10/2015.

Hale, A. R., Stoop, J. & Hommels, J. (1990). “Human error models as predictors of accident scenarios for designers in road transport systems”. In: *Ergonomics*, vol. 33, no. 10-11, pp. 1377–1387.

Heckhausen, H. & Gollwitzer, P. M. (June 1987). “Thought contents and cognitive functioning in motivational versus volitional states of mind”. In: *Motivation and Emotion*, vol. 11, no. 2, pp. 101–120.

Heckhausen, J. & Heckhausen, H. (2008). “Motivation and Action: Introduction and Overview”. In: *Motivation and Action*. Ed. by Heckhausen, J. & Heckhausen, H. Cambridge, USA: Cambridge University Press, pp. 1–9.

Heuer, H. (2015). “Fähigkeit (English Translation: Ability)”. In: *Dorsch Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/17/2014.

Heyes, M. P. & Ashworth, R. (Sept. 1972). “Further research on car-following models”. In: *Transportation Research*, vol. 6, no. 3, pp. 287–291.

Hidas, P. (Oct. 2002). “Modelling lane changing and merging in microscopic traffic simulation”. In: *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 5–6, pp. 351–371.

Hidas, P. (Feb. 2005). “Modelling vehicle interactions in microscopic simulation of merging and weaving”. In: *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 1, pp. 37–62.

Hirtz, P. (2003a). “Fähigkeit (English Translation: Abilitiy)”. In: *Sportwissenschaftliches Lexikon*. Ed. by Röthig, P. & Prohl, R. 7th. Schorndorf, Germany: Hofmann Publishing, pp. 188–189.

Hirtz, P. (2003b). “Fertigkeit (English Translation: Skill)”. In: *Sportwissenschaftliches Lexikon*. Ed. by Röthig, P. & Prohl, R. 7th. Schorndorf, Germany: Hofmann Publishing, pp. 196–197.

Hock, C. & Dickmanns, E. D. (1992). “Intelligent Navigation for Autonomous Robots Using Dynamic Vision”. In: *Congress of the International Society for Photogrammetry and Remote Sensing (ISPRS)*, pp. 900–915.

Hölzing, J. A. (2008). “Kano-Theorie der Kundenzufriedenheit (English Translation: Kano-Theory of Customer Satisfaction)”. In: *Die Kano-Theorie der Kundenzufriedenheitsmessung*. Wiesbaden, Germany: Gabler Verlag. Chap. 3, pp. 76–148.

Homeier, K. & Wolf, L. (Oct. 2011). "RoadGraph: High level sensor data fusion between objects and street network". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1380–1385.

Howard, R. A. (1980). "On Making Life and Death Decisions". In: *Societal Risk Assessment: How Safe is Safe Enough?* Ed. by Schwing, R. C. Boston, USA: Springer International Publishing, pp. 89–113.

Hundelshausen, F. von, Himmelsbach, M., Hecker, F., Mueller, A. & Wuensche, H.-J. (2008). "Driving with tentacles: Integral structures for sensing and motion". In: *Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673.

ISO (2010). *ISO/IEC/IEEE 24765: Systems and software engineering – Vocabulary*. Geneva, Switzerland: International Organization for Standardization (ISO).

ISO (2011). *ISO 26262:2011 Road vehicles - Functional safety*. Geneva, Switzerland: International Organization for Standardization (ISO).

Jarke, M., Bui, X. T. & Carroll, J. M. (Mar. 1998). "Scenario Management: An Interdisciplinary Approach". In: *Requirements Engineering*, vol. 3, no. 3-4, pp. 155–173.

Jo, K., Kim, J., Kim, D., Jang, C. & Sunwoo, M. (Aug. 2015). "Development of Autonomous Car - Part II: A Case Study on the Implementation of an Autonomous Driving System Based on Distributed Architecture". In: *IEEE Transactions on Industrial Electronics*, vol. 62, no. 8, pp. 5119–5132.

Jochem, T., Pomerleau, D. & Thorpe, C. (Sept. 1995). "Vision guided lane transition". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 30–35.

Jula, H., Kosmatopoulos, E. & Ioannou, P. (2000). "Collision avoidance analysis for lane changing and merging". In: *IEEE Transactions on Vehicular Technology*, vol. 49, no. 6, pp. 2295–2308.

Kaiser, F. (2014). "Verhalten (English Translation: Behavior)". In: *Dorsch Lexikon der Psychologie - Verlag Hans Huber*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/17/2014.

Kammel, S., Ziegler, J., Pitzer, B., Werling, M., Gindele, T., Jagzent, D., Schröder, J., Thuy, M., Goebl, M., Hundelshausen, F. von, Pink, O., Frese, C. & Stiller, C. (Sept. 2008). "Team AnnieWAY's autonomous system for the 2007 DARPA Urban Challenge". In: *Journal of Field Robotics*, vol. 25, no. 9, pp. 615–639.

Kanagaraj, V., Asaithambi, G., Kumar, C. H. N., Srinivasan, K. K. & Sivanandan, R. (Dec. 2013). "Evaluation of Different Vehicle Following Models Under Mixed Traffic Conditions". In: *Procedia - Social and Behavioral Sciences*, vol. 104, pp. 390–401.

Kanaris, A., Kosmatopoulos, E. & Loannou, P. (Nov. 2001). "Strategies and spacing requirements for lane changing and merging in automated highway systems". In: *IEEE Transactions on Vehicular Technology*, vol. 50, no. 6, pp. 1568–1581.

Kesting, A., Treiber, M. & Helbing, D. (2007). "General Lane-Changing Model MOBIL for Car-Following Models". In: *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1999, pp. 86–94.

Kesting, A., Treiber, M. & Helbing, D. (Oct. 2010). "Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity". In: *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1928, pp. 4585–4605.

Kita, H. & Fukuyama, K. (1999). "A merging-giveway behavior model considering interactions at expressway on-ramps". In: *Transportation and traffic theory*, vol. 14, pp. 173–187.

Kita, H., Tanimoto, K. & Fukuyama, K. (2002). "A game theoretical analysis of merging-giveway interaction: a joint estimation model". In: *Transportation and traffic theory*, vol. 15, pp. 503–518.

Kita, H. (Apr. 1999). "A merging–giveway interaction model of cars in a merging section: a game theoretic analysis". In: *Transportation Research Part A: Policy and Practice*, vol. 33, no. 3–4, pp. 305–312.

Kluckhohn, C. (1965). "Values and value orientation in the theory of action". In: *Toward a general theory of action*. Ed. by Parsons, T., Shils, E., Tolman, E. C. & Kluckhohn, C. Cambridge, USA: Harvard University Press.

Knaup, J. & Homeier, K. (Sept. 2010). "RoadGraph - Graph based environmental modelling and function independent situation analysis for driver assistance systems". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 428–432.

Knight, F. H. (Aug. 1964). *Risk, Uncertainty, and Profit*. Ed. by McClure, J. Re-issue edition. New York, USA: Sentry Press.

Kochenderfer, M. J. (2015). *Decision Making Under Uncertainty: Theory and Application*. Cambridge, USA: MIT Press.

Kometani, E. & Sasaki, T. (1958). "On the stability of traffic flow". In: *Journal of Operations Research*, vol. 2, no. 1, pp. 11–26.

Kopf, M. (1993). "Ein Beitrag zur modellbasierten, adaptiven Fahrerunterstützung für das Fahren auf deutschen Autobahnen (English Translation: A Contribution towards Model-Based, Adaptive Driver Assistance for Driving on German Highways)". Dissertation. Universität der Bundeswehr in München.

Krüger, W. (1992). *Situationsmodellierung in der Bildfolgenauswertung (English Translation: Situation Modeling in Image Sequence Evaluation)*. KI - Informatik Fachberichte. Berlin, Germany: Springer International Publishing.

Kujawski, C. (1995). "Deciding the behaviour of an autonomous mobile road vehicle". In: *International IFAC Conference on Intelligent Autonomous Vehicles*, pp. 404–409.

Kunz, F., Nuss, D., Wiest, J., Deusch, H., Reuter, S., Gritschneder, F., Scheel, A., Stubler, M., Bach, M., Hatzelmann, P., Wild, C. & Dietmayer, K. (June 2015). "Autonomous driving at Ulm University: A modular, robust, and sensor-independent fusion approach". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 666–673.

Kurniawati, H., Hsu, D. & Lee, W. S. (2008). "SARSOP : Efficient Point-Based POMDP Planning by Approximating Optimally Reachable Belief Spaces". In: *Proceedings Robotics: Science and systems (RSS)*. w/o page numbers.

Lawitzky, A., Althoff, D., Passenberg, C., Tanzmeister, G., Wollherr, D. & Buss, M. (June 2013). "Interactive scene prediction for automotive applications". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1028–1033.

Lee, J. H. (June 2011). "Model predictive control: Review of the three decades of development". In: *International Journal of Control, Automation and Systems*, vol. 9, no. 3, pp. 415–424.

Lenz, D., Kessler, T. & Knoll, A. (June 2016). "Tactical cooperative planning for autonomous highway driving using Monte-Carlo Tree Search". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 447–453.

Lewin, K. (1926). *Vorsatz, Wille und Bedürfnis (English Translation: Intent, volition, and need)*. Berlin, Germany: Springer International Publishing.

Liu, W., Kim, S.-W., Pendleton, S. & Ang, M. (June 2015). "Situation-aware decision making for autonomous driving on urban road using online POMDP". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1126–1133.

Luo, Y., Bai, H., Hsu, D. & Lee, W. S. (2016). "Importance Sampling for Online Planning under Uncertainty". In: *International Workshop on the Algorithmic Foundations of Robotics*, w/o page numbers.

Maciejowski, J. M., Visintini, A. L. & Lygeros, J. (Jan. 2007). "NMPC for Complex Stochastic Systems Using a Markov Chain Monte Carlo Approach". In: *Assessment and Future Directions of Nonlinear Model Predictive Control*. Ed. by Findeisen, R., Allgöwer, F. & Biegler, L. Lecture Notes in Control and Information Sciences. Berlin, Germany: Springer International Publishing, pp. 269–281.

Mackworth, A. K. (2011). "Architectures and ethics for robots: Constraint satisfaction as a unitary design framework". In: *Machine Ethics*. Ed. by Anderson, M. & Anderson, S. L. Cambridge, United Kingdom: Cambridge University Press, pp. 335–360.

Mählisch, M. (2009). "Filtersynthese zur simultanen Minimierung von Existenz-, Assoziations- und Zustandsunsicherheiten in der Fahrzeugumfelderfassung mit heterogenen Sensordaten (English Translation: Filter Synthesis for Simultaneous Minimization of Detection, Association, and State Uncertainties in Automotive Environment Perception with Heterogeneous Sensor Data)". Dissertation. Universität Ulm.

Mammar, S., Glaser, S. & Netto, M. (June 2006). "Time to line crossing for lane departure avoidance: a theoretical study and an experimental setting". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 2, pp. 226–241.

Matthaei, R. (2015). "Wahrnehmungsgestützte Lokalisierung in fahrstreifengenauen Karten für Fahrerassistenzsysteme und automatisches Fahren in urbaner Umgebung (English Translation: Perception-Supported Localization in Lane-Level Maps for Advanced Driver Assistance Systems and Automated Driving in Urban Environments)". Dissertation. TU Braunschweig.

Matthaei, R., Bagschik, G. & Maurer, M. (June 2014). "Map-relative localization in lane-level maps for ADAS and autonomous driving". In: *IEEE Intelligent Vehicles Symposium Proceedings*, pp. 49–55.

Matthaei, R. & Maurer, M. (2015). "Autonomous driving – a top-down-approach". In: *at - Automatisierungstechnik*, vol. 63, no. 3, pp. 155–167.

Matthaei, R., Reschka, A., Rieken, J., Dierkes, F., Ulbrich, S., Winkle, T. & Maurer, M. (2015). "Autonomous Driving". In: *Handbook of Driver Assistance Systems*. Ed. by Winner, H., Hakuli, S., Lotz, F. & Singer, C. London, United Kingdom: Springer International Publishing, pp. 1–31.

Maurer, M. (2000). "Flexible Automatisierung von Straßenfahrzeugen mit Rechnersehen (English Translation: Flexible Automation of Road Vehicles with Machine Vision)". Dissertation. Universität der Bundeswehr in München.

Maurer, M. (2015). "Einleitung (English Translation: Introduction)". In: *Autonomes Fahren*. Ed. by Maurer, M., Gerdes, C., Lenz, B. & Winner, H. Berlin, Germany: Springer International Publishing. Chap. 1, pp. 1–8.

Merriam-Webster (2015a). *Context*. Merriam-Webster Dictionary. URL: `http://www.merriam-webster.com/dictionary/context` (visited on 11/07/2015).

Merriam-Webster (2015b). *Intention*. Merriam-Webster Dictionary. URL: `http://www.merriam-webster.com/dictionary/intention` (visited on 11/07/2015).

Merriam-Webster (2015c). *Maneuver*. Merriam-Webster Dictionary. URL: `http://www.merriam-webster.com/dictionary/maneuver` (visited on 11/07/2015).

Merriam-Webster (2015d). *Utility*. Merriam-Webster Dictionary. URL: `http://www.merriam-webster.com/dictionary/utility` (visited on 11/07/2015).

Meyer (1977). "Situation". In: *Meyers enzyklopädisches Lexikon*. Mannheim, Germany: Bibliographisches Institut AG, vol. 21.

Michon, J. A. (1985). "A Critical View of Driver Behavior Models: What Do We Know, What Should We Do?" In: *Human Behavior and Traffic Safety*. Ed. by Evans, L. & Schwing, R. C. New York, USA: Springer International Publishing, pp. 485–524.

Miura, J., Negishi, Y. & Shirai, Y. (2002). "Mobile robot map generation by integrating omnidirectional stereo and laser range finder". In: *International IEEE Conference on Intelligent Robots and System*. Vol. 1, pp. 250–255.

Mock-Hecker, R. (1994). *Wissensbasierte Erkennung kritischer Verkehrssituationen: Erkennung von Plankonflikten (English Translation: Knowledge-Based Detection of Critical Traffic Situations: Detection of Plan Conflicts)*. Düsseldorf, Germany: VDI Verlag.

Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., Haehnel, D., Hilden, T., Hoffmann, G., Huhnke, B., Johnston, D., Klumpp, S., Langer, D., Levandowski, A., Levinson, J., Marcil, J., Orenstein, D., Paefgen, J., Penny, I., Petrovskaya, A., Pflueger, M., Stanek, G., Stavens, D., Vogt, A. & Thrun, S. (2008). "Junior: The stanford entry in the urban challenge". In: *Journal of Field Robotics*, vol. 25, no. 9, pp. 569–597.

Moridpour, S., Sarvi, M. & Rose, G. (July 2010). "Lane changing models: a critical review". In: *Transportation Letters*, vol. 2, no. 3, pp. 157–173.

Muigg, A. (2009). "Implizites Workloadmanagement - Konzept einer zeitlich-situativen Informationsfilterung im Automobil (English Translation: Implicit Workload Management - Concept of a Temporal-Situative Information Filtering in an Automobile)". Dissertation. TU München.

Mukai, M. & Kawabe, T. (Oct. 2006). "Model predictive control for lane change decision assist system using hybrid system representation". In: *International Joint SICE-ICASE Conference*, pp. 5120–5125.

Muller, T. & Zuylen, H. van (2006). "A Driving Behavior Model for Microscopic Online Simulation based on Remote Sensing and Equipped Vehicle Data". In: *Transportation Research Board Annual Meeting*. w/o page numbers.

Munos, R. & Moore, A. (Nov. 2002). "Variable Resolution Discretization in Optimal Control". In: *Machine Learning*, vol. 49, no. 2-3, pp. 291–323.

Murray, D. & Little, J. J. (2000). "Using Real-Time Stereo Vision for Mobile Robot Navigation". In: *Autonomous Robots*, vol. 8, no. 2, pp. 161–171.

Naranjo, J., Gonzalez, C., Garcia, R. & Pedro, T. de (Sept. 2008). "Lane-Change Fuzzy Control in Autonomous Vehicles for the Overtaking Maneuver". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 438–450.

Nau, R. F. & McCardle, K. F. (Apr. 1990). "Coherent behavior in noncooperative games". In: *Journal of Economic Theory*, vol. 50, no. 2, pp. 424–444.

NHTSA (2013). *U.S. Department of Transportation Releases Policy on Automated Vehicle Development*. URL: `http://www.nhtsa.gov/About+NHTSA/Press+Releases/U.S.+Department+of+Transportation+Releases+Policy+on+Automated+Vehicle+Development`.

NHTSA (2016). *Federal Automated Vehicles Policy*. U.S. Department of Transportation.

Nilsson, J., Ali, M., Falcone, P. & Sjöberg, J. (Oct. 2013). "Predictive manoeuvre generation for automated driving". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 418–423.

Nothdurft, T. (2014). "Ein Kontextmodell für sicherheitsrelevante Anwendungen in der autonomen Fahrzeugführung (English Title: A Context Model for Safety-Relevant Applications in Automated Vehicle Guidance)". Dissertation. TU Braunschweig.

Nothdurft, T., Hecker, P., Frankiewicz, T., Gacnik, J. & Köster, F. (Sept. 2011). "Reliable Information Aggregation and Exchange for Autonomous Vehicles". In: *IEEE Vehicular Technology Conference (VTC)*, pp. 1–5.

Oberlander, J., Uhl, K., Zollner, J. M. & Dillmann, R. (May 2008). "A region-based SLAM algorithm capturing metric, topological, and semantic properties". In: *International IEEE Conference on Robotics and Automation (ICRA)*, pp. 1886–1891.

Oxford (2015a). *Behavior*. Oxford Dictionaries. URL: `http://www.oxforddictionaries.com/definition/american_english/behavior` (visited on 07/11/2015).

Oxford (2015b). *Harm*. Oxford Dictionaries. URL: `http://www.oxforddictionaries.com/definition/english/harm` (visited on 07/11/2015).

Oxford (2015c). *Maneuver*. Oxford Dictionaries. URL: `http://www.oxforddictionaries.com/definition/american_english/maneuver` (visited on 07/11/2015).

Oxford (2015d). *Scenario*. Oxford Dictionaries. URL: `http://www.oxforddictionaries.com/definition/english/scenario` (visited on 07/11/2015).

Oxford (2017a). *Map*. Oxford Dictionaries. URL: `https://en.oxforddictionaries.com/definition/map` (visited on 02/10/2017).

Oxford (2017b). *Systemic*. Oxford Dictionaries. URL: `https://en.oxforddictionaries.com/definition/systemic`.

Pascheka, P. & During, M. (Sept. 2015). "Advanced cooperative decentralized decision making using a cooperative reward system". In: *International Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, pp. 1–7.

Patz, B. J., Papelis, Y., Pillat, R., Stein, G. & Harper, D. (Aug. 2008). "A practical approach to robotic design for the DARPA Urban Challenge". In: *Journal of Field Robotics*, vol. 25, no. 8, pp. 528–566.

Pellkofer, M. & Dickmanns, E. (2002). "Behavior decision in autonomous vehicles". In: *IEEE Intelligent Vehicles Symposium (IV)*. Vol. 2, pp. 495–500.

Pellkofer, M. (2003). "Verhaltensentscheidung für autonome Fahrzeuge mit Blickrichtungssteuerung (English Translation: Behavior Decision Making for Autonomous Vehicles with Gaze Control)". Dissertation. Universität der Bundeswehr in München.

Pineau, J., Gordon, G. & Thrun, S. (2003). “Point-based value iteration : An anytime algorithm for POMDPs”. In: *Proceedings International Joint Conference on Artificial Intelligence (IJCAI)*. Ed. by Gottlob, G. & Walsh, T., pp. 1025–1032.

Powers, T. (July 2006). “Prospects for a Kantian Machine”. In: *IEEE Intelligent Systems*, vol. 21, no. 4, pp. 46–51.

Puca, R. M. (2015). “Intention”. In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Puca, R. M. (2016). “Motiv (English Translation: Motive)”. In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Rahman, M., Chowdhury, M., Xie, Y. & He, Y. (Dec. 2013). “Review of Microscopic Lane-Changing Models and Future Research Opportunities”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1942–1956.

Rao, A. S. R. & Georgeff, M. P. (1991). “Modeling Rational Agents within a BDI-Architecture”. In: *International Conference on Principles of Knowledge Representation and Reasoning*. Ed. by Allen, J., Fikes, R. & Sandewall, E., pp. 473–484.

Rasmussen, J. (May 1983). “Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models”. In: *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-13, no. 3, pp. 257–266.

Rauskolb, F., Berger, K., Lipski, C., Magnor, M., Cornelsen, K., Effertz, J., Form, T., Graefe, F., Ohl, S., Doering, M., Homeier, K., Morgenroth, J., Wolf, L., Basarke, C. & Berger, C. (2008). “Caroline: An Autonomously Driving Vehicle for Urban Environments”. In: *Journal of Field Robotics*, vol. 25, no. 9, pp. 674–724.

Reichardt, D. (1996). “Kontinuierliche Verhaltenssteuerung eines autonomen Fahrzeugs in dynamischer Umgebung (English Translation: Continuous Behavior Control of an Autonomous Vehicle in a Dynamic Environment)”. Dissertation. Universität Kaiserslautern.

Reichel, M. (Apr. 2013). “Situationsanalyse für fortschrittliche Fahrerassistenzsysteme (English Translation: Situation Analysis for Advanced Driver Assistance Systems)”. Dissertation. TU Braunschweig.

Reichel, M., Botsch, M., Rauschecker, R., Siedersberger, K. & Maurer, M. (2010). “Situation aspect modelling and classification using the Scenario Based Random Forest algorithm for convoy merging situations”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 360–366.

Reschka, A. (2015). *Personal communication regarding autonomous driving and automated driving.*

Reschka, A. (2017). “Fertigkeiten- und Fähigkeitengraphen als Grundlage des sicheren Betriebs von automatisierten Fahrzeugen im öffentlichen Straßenverkehr in städtischer Umgebung (English Translation: Skill and Ability Graphs as a Foundation for Safe Operations of Automated Vehicles in Public Road Traffic in Urban Environments)”. Dissertation, forthcoming. TU Braunschweig.

Reschka, A., Bagschik, G., Ulbrich, S., Nolte, M. & Maurer, M. (2015). “Ability and skill graphs for system modeling, online monitoring, and decision support for vehicle guidance systems”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 933–939.

Reschka, A., Böhmer, J. R., Gacnik, J., Köster, F., Wille, J. M. & Maurer, M. (2011). “Development of Software for Open Autonomous Automotive Systems in the Stadtpilot-Project”. In: *International Workshop on Intelligent Transportation (WIT)*, pp. 81–86.

Reschka, A. & Maurer, M. (2015). “Conditions for a safe state of automated road vehicles”. In: *Information Technology*, vol. 57, no. 4, pp. 215–222.

Rheinberg, F. (1989). *Zweck und Tätigkeit: Motivationspsychologische Analysen zur Handlungsveranlassung (English Translation: Purpose and Activity: Motivation Psychological Analysis for Activity Inducement)*. Göttingen, Germany: Hogrefe Verlag für Psychologie.

Richardson, M. & Domingos, P. (Jan. 2006). “Markov logic networks”. In: *Machine Learning*, vol. 62, no. 1-2, pp. 107–136.

Rieken, J., Matthaei, R. & Maurer, M. (Sept. 2015). “Toward Perception-Driven Urban Environment Modeling for Automated Road Vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 731–738.

Riemersma, J. B. J. (June 1979). “Perception in traffic”. In: *Urban Ecology*, vol. 4, no. 2, pp. 139–149.

Roelofsen, M., Bie, J., Jin, L. & Arem, B. van (June 2010). “Assessment of safety levels and an innovative design for the Lane Change Assistant”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 83–88.

Rorbech, J. (1974). *The Multi-Lane Traffic Flow Process*. Copenhagen, Denmark: Ministry of Public Works, Denmark.

Ross, S., Pineau, J., Paquet, S. & Chaib-Draa, B. (July 2008). “Online Planning Algorithms for POMDPs”. In: *The journal of artificial intelligence research*, vol. 32, no. 2, pp. 663–704.

Rössler, B. (2010). *Situation Analysis and Adaptive Risk Assessment for Advanced Intersection Safety Systems*. Berlin, Germany: Logos-Verlag.

Rothengatter, T. (1988). “Risk and the absence of pleasure: a motivational approach to modelling road user behaviour”. In: *Ergonomics*, vol. 31, no. 4, pp. 599–607.

Ruf, M., Ziehn, J. R., Rosenhahn, B. & Al, E. (2014a). “Situation prediction and reaction control (SPARC): Patent pending”. In: *Workshop Fahrerassistenzsysteme (FAS 2014)*, pp. 55–66.

Ruf, M., Ziehn, J. R., Willersinn, D., Rosenhahn, B., Beyerer, J. & Gotzig, H. (Dec. 2014b). “A Continuous Approach to Autonomous Driving”. In: *Proceedings of the Conference on Vehicle and Infrastructure Safety Improvement in Adverse Conditions and Night Driving (VISION 2014)*.

Russell, S. & Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*. 3rd edition. Upper Saddle River, New Jersey: Prentice Hall.

SAE (2016). *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. Warrendale, USA: SAE International.

Saust, F., Bley, O., Kutzner, R., Wille, J. M., Friedrich, B. & Maurer, M. (Sept. 2010). “Exploitability of vehicle related sensor data in cooperative systems”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1724–1729.

Saust, F., Wille, J. M. & Maurer, M. (May 2012). “Energy-Optimized Driving with an Autonomous Vehicle in Urban Environments”. In: *IEEE Vehicular Technology Conference (VTC Spring)*, pp. 1–5.

Schakel, W. J., Knoop, V. L. & Arem, B. van (2012). “LMRS: An Integrated Lane Change Model with Relaxation and Synchronization”. In: *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2316, pp. 47–57.

Schildbach, G. & Borrelli, F. (June 2015). “Scenario model predictive control for lane change assistance on highways”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 611–616.

Schmidt, M. T., Hoffmann, U. & Bouzouraa, M. E. (2014). “A Novel Goal Oriented Concept for Situation Representation for ADAS and Automated Driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 886–893.

Schröder, J. (Sept. 2009). *Adaptive Verhaltensentscheidung und Bahnplanung für kognitive Automobile (English Translation: Adaptive Behavior Decision Making and Path Planning for Cognitive Vehicles)*. Karlsruhe, Germany: KIT Scientific Publishing.

Schubert, R. (Mar. 2012). “Evaluating the Utility of Driving: Toward Automated Decision Making Under Uncertainty”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 354–364.

Schubert, R. & Wanielik, G. (2011). “A Unified Bayesian Approach for Object and Situation Assessment”. In: *IEEE Intelligent Transportation Systems Magazine*, vol. 3, no. 2, pp. 6–19.

Schubert, R. (2011). “Integrated Bayesian object and situation assessment for lane change assistance”. Dissertation. TU Chemnitz.

Schubert, R., Schulze, K. & Wanielik, G. (Sept. 2010). “Situation Assessment for Automatic Lane-Change Maneuvers”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 607–616.

Schuldt, F. (2017). “Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen (English Translation: A Contribution towards Methological Testing of Automated Driving Functions with Virtual Environments)”. Dissertation. TU Braunschweig, forthcoming.

Schwarting, W. & Pascheka, P. (Oct. 2014). “Recursive conflict resolution for cooperative motion planning in dynamic highway traffic”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1039–1044.

Seitz, D. (2015). “Situation Awareness”. In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Shani, G., Pineau, J. & Kaplow, R. (July 2013). “A survey of point-based POMDP solvers”. In: *Autonomous Agents and Multi-Agent Systems*, vol. 27, no. 1, pp. 1–51.

Shen, J. & Jin, X. (Sept. 2012). “Detailed traffic animation for urban road networks”. In: *Graphical Models*, vol. 74, no. 5, pp. 265–282.

Li-sheng, J., Wen-ping, F., Ying-nan, Z., Shuang-bin, Y. & Hai-jing, H. (June 2009). “Research on safety lane change model of driver assistant system on highway”. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1051–1056.

Shirazi, M. & Morris, B. (June 2015). “Observing behaviors at intersections: A review of recent studies and developments”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1258–1263.

Siedersberger, K.-H. (2003). “Komponenten zur automatischen Fahrzeugführung in sehenden (semi-)autonomen Fahrzeugen (English Translation: Components for Automated Vehicle Guidance in Perceiving (Semi)-Autonomous Vehicles)”. Dissertation. Universität der Bundeswehr in München.

Silver, D. & Veness, J. (2010). “Monte-Carlo Planning in Large POMDPs”. In: *Neural Information Processing Systems*, pp. 2164–2172.

Sivaraman, S., Morris, B. & Trivedi, M. (Oct. 2013). “Observing on-road vehicle behavior: Issues, approaches, and perspectives”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1772–1777.

Sivaraman, S. & Trivedi, M. (Oct. 2014). “Dynamic Probabilistic Drivability Maps for Lane Change and Merge Driver Assistance”. In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2063–2073.

Six, B. (2015). “Werte (English Translation: Values)”. In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/07/2015.

Skog, I. & Händel, P. (2012). “State-of-the-Art In-Car Navigation: An Overview”. In: *Handbook of Intelligent Vehicles*. Ed. by Eskandarian, A. London, United Kingdom: Springer International Publishing, pp. 435–462.

Somani, A., Ye, N., Hsu, D. & Lee, W. S. (2013). "DESPOT: Online POMDP Planning with Regularization". In: *Conference on Neural Information Processing Systems (NIPS)*, pp. 1772–1780.

Spaan, M. T. J. & Vlassis, N. (2005). "Perseus : Randomized Point-based Value Iteration for POMDPs". In: *Journal of Artificial Intelligence Research*, vol. 24, pp. 195–220.

Sparmann, U. (1978). "Spurwechselvorgänge auf zweispurigen BAB Richtungsfahrbahnen (English Translation: Lane Changes on Two-Lane BAB Carriageways)". Dissertation. Universität Karlsruhe (TH).

Spieß, E. (2014). "Kooperation (English Translation: Cooperation)". In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/17/2014.

Stiller, C., Burgard, W., Deml, B., Eckstein, L., Flemisch, F., Köster, F., Maurer, M. & Wanielik, G. (2013). *Kooperativ interagierende Automobile (English Title: Cooperatively Interacting Automobiles)*. Deutsche Forschungsgemeinschaft (DFG). URL: `http://www.mrt.kit.edu/spp1835/download/spp1835.pdf`.

Stolte, T., Reschka, A., Bagschik, G. & Maurer, M. (Sept. 2015). "Towards Automated Driving: Unmanned Protective Vehicle for Highway Hard Shoulder Road Works". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 672–677.

Strang, T. & Linnhoff-Popien, C. (2004). "A Context Modeling Survey". In: *Workshop on Advanced Context Modelling, Reasoning and Management at the International Conference on Ubiquitous Computing (UbiComp)*, w/o page numbers.

StVO (2013). *Straßenverkehrs-Ordnung (StVO)*. URL: `http://www.gesetze-im-internet.de/stvo_2013/BJNR036710013.html`.

Tas, O. S., Kuhnt, F., Zollner, J. M. & Stiller, C. (June 2016). "Functional system architectures towards fully automated driving". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 304–309.

Tetlock, P. E. (June 2000). "Cognitive Biases and Organizational Correctives: Do Both Disease and Cure Depend on the Politics of the Beholder?" In: *Administrative Science Quarterly*, vol. 45, no. 2, pp. 293–326.

Tetlock, P. E. (July 2003). "Thinking the unthinkable: sacred values and taboo cognitions". In: *Trends in Cognitive Sciences*, vol. 7, no. 7, pp. 320–324.

Thomason, M. G. & Gonzalez, R. C. (1985). "Data Structures and Databases in Digital Scene Analysis". In: *Advances in Information Systems Science*. Ed. by Tou, J. T. New York, USA: Springer International Publishing, pp. 1–47.

Thornton, S. M., Pan, S., Erlien, S. M. & Gerdes, J. C. (2016). "Incorporating Ethical Considerations Into Automated Vehicle Control". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–11.

Thrun, S., Burgard, W. & Fox, D. (2005). *Probabilistic Robotics*. 2nd edition. Cambridge, USA: MIT Press.

Todosiev, E. P. (1963). "The action point model of the driver-vehicle system". Dissertation. Ohio State University.

Toledo, T. (2003). "Integrated Driving Behavior Modeling". Dissertation. Massachusetts Institute of Technology.

Toledo, T., Koutsopoulos, H. & Ben-Akiva, M. (Jan. 2003). "Modeling Integrated Lane-Changing Behavior". In: *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1857, pp. 30–38.

TRB (Dec. 2000). *Highway Capacity Manual: 2000*. 4th edition. Washington, DC: Transportation Research Board.

Treiber, M. & Helbing, D. (2001). "Microsimulations of Freeway Traffic Including Control Measures". In: *at - Automatisierungstechnik*, vol. 49, no. 11/2001, p. 478.

Treiterer, J. & Myers, J. (1974). "The hysteresis phenomenon in traffic flow". In: *Transportation and traffic theory*, vol. 6, pp. 13–38.

Ulbrich, S. & Maurer, M. (2013). "Probabilistic online POMDP decision making for lane changes in fully automated driving". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 2063–2067.

Ulbrich, S., Nothdurft, T., Maurer, M. & Hecker, P. (June 2014). "Graph-based context representation, environment modeling and information aggregation for automated driving". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 541–547.

Ulbrich, S. (2011). "Intelligent Decision Making and Maneuver Planning for Autonomous Driving in Urban Traffic Environments". Diploma-Thesis. TU Braunschweig.

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015a). "Ermitteln von Fahrstreifenflussgeschwindigkeiten zur Bewertung des Nutzens von Fahrstreifenwechseln in dünnem Verkehr (English Translation: Determining Lane Specific Traffic Flow Velocities to Evaluate the Benefit of Performing Lane Changes in Sparse Traffic)". Disclosure of Invention (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015b). "Kommunikation einer Fahrstreifenwechselabsicht (English Translation: Communication of a Lane Change Intention)". Patent DE 102015219934 A1 (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015c). "Marginale, integrale Vorteilhaftigkeit bei der Bewertung des Überholnutzens eines geringfügig langsameren Vorderfahrzeugs (English Translation: Marginal, Integral Benefit for Evaluating the Overtaking Utility of a Marginally Slower Front Vehicle)". Pending Patent Application (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015d). “Verfahren und Vorrichtung in einem Kraftfahrzeug zum automatisierten Fahren (English Translation: Method and Device in a Motor Vehicle for an Automatic Drive)”. Patent DE 102015217498 A1 (Germany).

Ulbrich, S., Großjohann, S., Appelt, C. & Homeier, K. (2015e). “Verfahren zur Befolgung des Rechtsfahrgebots bei der Bewertung des Nutzens von Fahrstreifenwechseln (English Translation: Procedure for Adherence of the Right Lane Driving Order Rule in the Benefit Evaluation of Lane Changes)”. Pending Patent Application (Germany).

Ulbrich, S., Großjohann, S., Appelt, C., Homeier, K., Rieken, J. & Maurer, M. (2015f). “Structuring Cooperative Behavior Planning Implementations for Automated Driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 2159–2165.

Ulbrich, S. & Maurer, M. (2014). “Evaluation einer taktischen Verhaltensentscheidungsfindung für Fahrstreifenwechsel beim vollautomatisierten Fahren in Städten (English Translation: Evaluation of a Tactical Behavior Decision Making Algorithm for Lane Changes in Fully Automated Driving in Cities)”. In: *Workshop Fahrerassistenzsysteme (FAS 2014)*, pp. 147–158.

Ulbrich, S. & Maurer, M. (2015a). “Situation Assessment in Tactical Lane Change Behavior Planning for Automated Vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 975–981.

Ulbrich, S. & Maurer, M. (2015b). “Towards Tactical Lane Change Behavior Planning for Automated Vehicles”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 989–995.

Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F. & Maurer, M. (2015g). “Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving”. In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 982–988.

Ulbrich, S., Menzel, T., Reschka, A., Schuldt, F. & Maurer, M. (2015h). “Definition der Begriffe Szene, Situation und Szenario für das automatisierte Fahren (English Translation: Defining the Terms Scene, Situation and Scenario for Automated Driving)”. In: *Workshop Fahrerassistenzsysteme (FAS 2015)*, 105–117, German Version of Ulbrich et al. (2015g).

Ulbrich, S., Reschka, A., Kochenderfer, M. & Maurer, M. (2017a). “Towards Value-Oriented Behavior Planning”. In: *Transactions of Intelligent Transportation Systems*, forthcoming.

Ulbrich, S., Reschka, A., Rieken, J., Ernst, S., Bagschik, G., Dierkes, F., Nolte, M. & Maurer, M. (2017b). *Towards a Functional System Architecture for Automated Vehicles*. URL: `https://arxiv.org/abs/1703.08557`.

Ulbrich, S., Schuldt, F., Homeier, K., Steinhoff, M., Menzel, T., Krause, J. & Maurer, M. (2017c). "Testing and Validating Tactical Lane Change Behavior Planning for Automated Driving". In: *Automated Driving - Safer and more efficient future driving*. Ed. by Horn, M. & Watzenig, D. Springer International Publishing AG, pp. 451–471.

Vacek, S. (2009). "Videogestützte Umfelderfassung zur Interpretation von Verkehrssituationen für kognitive Automobile (English Translation: Video-Supported Environment Perception for Interpreting Traffic Situations for Cognitive Automobiles)". Dissertation. Universität Karlsruhe (TH).

VDA (2015). *Automatisierung - Von Fahrerassistenzsystemen zum automatisierten Fahren (English Translation: Automation - from advanced driver assistance systems towards automated driving)*. Berlin, Germany: Verband der Automobilindustrie e.V.

Veruggio, G. & Operto, F. (2008). "Roboethics: Social and Ethical Implications of Robotics". In: *Springer Handbook of Robotics*. Ed. by Siciliano, B. & Khatib, O. Springer, pp. 1499–1524.

Von Benda, H. (June 1985). *Die Häufigkeit von Verkehrssituationen (English Translation: The Frequency of Traffic Situations)*. Bergisch Gladbach, Germany: Bundesanstalt für Straßenwesen (BASt).

Wakasugi, T. (2005). "A Study on Warning Timing for Lane Change Decision Aid Systems Based on Driver's Lane Change Maneuver". In: *International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, w/o page numbers.

Wang, F., Yang, M. & Yang, R. (June 2009). "Conflict-Probability-Estimation-Based Overtaking for Intelligent Vehicles". In: *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 366–370.

Wang, H., Mao, T., Kang, X. & Wang, Z. (2014). "An all-in-one efficient lane-changing model for virtual traffic". In: *Computer Animation and Virtual Worlds*, vol. 25, no. 3-4, pp. 383–391.

Wang, J., Liu, R. & Montgomery, F. (2005). "A Simulation Model for Motorway Merging Behaviour". In: *International Symposium on Transportation and Traffic Theory*, pp. 281–301.

Wang, M. (Oct. 2014). "Generic Model Predictive Control Framework for Advanced Driver Assistance Systems". Dissertation. TU Delft.

Watzlawick, P., Bavelas, J. B. & Jackson, D. D. (1967). *Pragmatics of human communication: a study of interactional patterns, pathologies, and paradoxes*. New York, USA: Norton.

Webster, N. A. (2007). "Driver Model for Traffic Simulation, with Tactical Lane Changing Behavior". Dissertation. University of Toronto.

Wei, J., Dolan, J. M. & Litkouhi, B. (June 2010). "A prediction- and cost function-based algorithm for robust autonomous freeway driving". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 512–517.

Wei, J., Dolan, J. M., Snider, J. M. & Litkouhi, B. (2011). "A point-based MDP for robust single-lane autonomous driving behavior under uncertainties". In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2586–2592.

Wei, J., Snider, J. M., Gu, T., Dolan, J. M. & Litkouhi, B. (June 2014). "A behavioral planning framework for autonomous driving". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 458–464.

Weißel, F. (2009). "Stochastische modell-prädiktive Regelung nichtlinearer Systeme (English Translation: Stochastic Model Predictive Control of Nonlinear Systems)". Dissertation. Universität Karlsruhe (TH).

Werling, M., Gindele, T., Jagszent, D. & Gröll, L. (June 2008). "A robust algorithm for handling moving traffic in urban scenarios". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1108–1112.

Werling, M., Kammel, S., Ziegler, J. & Groll, L. (Dec. 2011). "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds". In: *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359.

Werling, M. (2010). "Ein neues Konzept für die Trajektoriengenerierung und -stabilisierung in zeitkritischen Verkehrsszenarien (English Translation: A Novel Concept for Trajectory Generation and Stabilization in Time-Critical Traffic Scenarios)". Dissertation. Karlsruher Institut für Technologie.

Werling, M., Ziegler, J., Kammel, S. & Thrun, S. (2010). "Optimal trajectory generation for dynamic street scenarios in a Frenét Frame". In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 987–993.

Wershofen, K. P. & Graefe, V. (1996). "Situationserkennung als Grundlage der Verhaltenssteuerung eines mobilen Roboters (English Translation: Situation Recognition as a Base for Behavior Control of a Mobile Robot)". In: *Fachgespräch Autonome Mobile Systeme*, pp. 170–179.

Wiedemann, R. (1974). "Simulation des Straßenverkehrsflusses (English Translation: Simulation of Road Traffic Flow)". Dissertation. Karlsruhe Institute of Technology.

Wille, J. M. (2012). "Manöverübergreifende autonome Fahrzeugführung in innerstädtischen Szenarien am Beispiel des Stadtpilotprojekts (English Title: Maneuver-Comprehensive Autonomous Vehicle Guidance in Urban Scenarios in the Example of the Stadtpilot Project)". Dissertation. TU Braunschweig.

Wille, J. M., Saust, F. & Maurer, M. (Sept. 2010). "Comprehensive treated sections in a trajectory planner for realizing autonomous driving in Braunschweig's urban traffic". In: *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 647–652.

Williams, B. (Dec. 1981). "Persons, Character and Morality". In: *Moral Luck: Philosophical Papers 1973–1980*. Cambridge, United Kingdom: Cambridge University Press. Chap. 1, pp. 1–19.

Winner, H., Geyer, S. & Sefati, M. (2013). "Maße für den Sicherheitsgewinn von Fahrerassistenzsystemen (English Translation: Metrics for Safety Gain of Advanced Driver Assistance Systems)". In: *Darmstädter Kolloquium Mensch + Fahrzeug*, pp. 63–82.

Wirtz, M. A. (2015a). "Kontext (English Translation: Context)". In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Wirtz, M. A. (2015b). "Risiko (English Translation: Risk)". In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Wirtz, M. A. (2015c). "Situation". In: *Dorsch - Lexikon der Psychologie*. Ed. by Wirtz, M. A. Bern, Switzerland: Verlag Hans Huber, accessed online on 11/25/2015.

Wooldridge, M. (1999). "Intelligent agents". In: *Multiagent Systems - A Modern Approach To Distributed Artificial Intelligence*. Ed. by Weiss, G. Cambridge, USA: MIT Press, pp. 27–78.

Ye, N., Au, N. Y. E., Somani, A. & Hsu, D. (2017). "DESPOT: Online POMDP Planning with Regularization". In: *Journal of Artificial Intelligence Research*, vol. 58, pp. 231–266.

Zapp, A. (1988). "Automatische Straßenfahrzeugführung durch Rechnersehen (English Translation: Automated Road Vehicle Guidance by Computer Vision)". Dissertation. Universität der Bundeswehr in München.

Ziegler, J., Bender, P., Schreiber, M., Lategahn, H., Strauss, T., Stiller, C., Dang, T., Franke, U., Appenrodt, N., Keller, C., Kaus, E., Herrtwich, R., Rabe, C., Pfeiffer, D., Lindner, F., Stein, F., Erbs, F., Enzweiler, M., Knöppel, C., Hipp, J., Haueis, M., Trepte, M., Brenk, C., Tamke, A., Ghanaat, M., Braun, M., Joos, A., Fritz, H., Mock, H., Hein, M. & Zeeb, E. (2014a). "Making Bertha Drive - An Autonomous Journey on a Historic Route". In: *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20.

Ziegler, J., Bender, P., Lategahn, H., Schreiber, M., Strauss, T., Dang, T. & Stiller, C. (2014b). "Kartengestütztes automatisiertes Fahren auf der Bertha-Benz-Route von Mannheim nach Pforzheim (English Translation: Map-Based, Automated Driving on the Bertha-Benz-Route from Mannheim to Pforzheim)". In: *Workshop Fahrerassistenzsysteme (FAS)*, pp. 79–93.

Ziehn, J., Ruf, M., Rosenhahn, B., Willersinn, D., Beyerer, J. & Gotzig, H. (June 2015). "Correspondence between variational methods and Hidden Markov Models". In: *IEEE Intelligent Vehicles Symposium (IV)*, pp. 380–385.